The Playthrough Evaluation Framework

Reliable Usability Evaluation for Video Games

Gareth R. White

Submitted for the degree of Ph.D.

Department of Informatics, University of Sussex

Monday 10th November, 2014

Declaration

I hereby declare that this thesis has not been submitted, either in the same or different form, to this or any other university for a degree.

Signature:

Acknowledgements

Running the Brighton marathon in 2010 was a good warm up for completing this thesis. It was during that experience that I realised how important support was, whether it was from friends on Facebook *like*'ing my training updates, people donating to my charity, family and total strangers completely putting their hearts into cheering, and the camaraderie of peers going through the ordeal together. Similarly this thesis genuinely could not have been completed without the support of a host of people, just some of whom are listed here,

Supervisors: Judith Good, Geraldine Fitzpatrick, Graham McAllister.

Committee: Peter Cheng, Ben du Boulay, Pablo Romero.

Colleagues: Pejman Mirza-babaei, Emma Foley, Seb Long; Anna Jordanous, Chris Kiefer, Dave Harley, Layda Gognora, Manuela Jungmann, Raphael Commins; Eric Harris, Hilary Smith, Jon Rimmer, Katy Howland, Lesley Axelrod, Madeline Balaam.

Family: Mum & Dad, Matt & Emma, Grandad.

Dear friends: Ken MacKriell, Paul Sumpner, Phil Harris

Team Kimberley: Cliff Dargonne, and too many housemates to name.

London K-pop community: 사랑해!

Game Ph.D. inspiration: Babsie Lippe.

Spiritual support: Thai Forest Sangha, Bodhi Tree Brighton, Triratna Buddhist Community.

Arahaṃ sammāsambuddho bhagavā, buddhaṃ bhagavantaṃ abhivādemi
Svākkhāto bhagavatā dhammo, dhammaṃ namassāmi
Supaṭipanno bhagavato sāvakasaṅgho, saṅghaṃ namāmi.

Preface

This thesis developed from personal experience working as a video game programmer between 1998 and 2006, for three different studios in three different countries. During that time I worked on several different projects, from small to large, including one which sold over two million units and generated an estimated one hundred million dollars in revenue. However, for each of these great success stories in the industry there are countless failures. Vast sums of money are lost, as well as the personal investment of time and effort from tens to hundreds of talented individuals. This experience was the primary motivation to return to academia, with my personal research question being, "To what extent can academic study address the problems faced in the video game industry?".

There are many interesting questions to ask about video games, many problems to solve. While some of the more exciting and popular areas of research deal with intangible qualities such as understanding fun, there are significant foundations missing in the field. In this early period of academic research, it is important to develop a solid basis on which future work can stand. Work that does not stand on a solid grounding in the basics may appear to be glamorous, but is like the proverbial house built on sand.

Research Context

This thesis was made possible by funding provided by the University of Sussex's Graduate Teaching Assistantship programme. A three year studentship stipend was awarded to fund a Ph.D. in the area of video game usability and user experience, within a framework of human-computer interaction.

During this period the work was conducted within the Interact lab[1], in the Human Centred Technology team[2], part of the Digital Interactive Systems research group[3]. Other work in the Interact lab includes research into technology assisted accessibility, pedagogy, and health, and is conducted with a human-computer interaction perspective.

Early research conducted at the beginning of this Ph.D. was more focussed on the use of interactive technologies for the elderly. During this stage it became apparent that there were fundamental issues with usability evaluation of video games in general that needed to be addressed, regardless of the player demographic involved. These observations informed a change of focus but influenced the work in this thesis, and so the research agenda moved to

[1] http://www.informatics.sussex.ac.uk/research/groups/interact/
[2] http://www.informatics.sussex.ac.uk/research/groups/hct/
[3] http://www.sussex.ac.uk/intsys/

these more general concerns of evaluation methodologies themselves.

In addition, during this time the University of Sussex provided substantial funding for the creation of a commercial video game usability evaluation studio, Vertical Slice. In my role as Director of Games Research, along with my colleague Dr. Graham McAllister, I co-founded and ran this startup company, providing professional evaluation services to the games industry internationally. Our user testing and expert evaluation studies are credited in high profile games that have received critical and commercial success, including the top ten, award winning first-person shooter game *Crysis 2*. This practical experience proved invaluable for understanding the issues involved in conducting testing and evaluation for games in the real world, and influenced this thesis in innumerable ways.

The Playthrough Evaluation Framework

Reliable Usability Evaluation for Video Games

Gareth R. White

Abstract

This thesis presents the playthrough evaluation framework, a novel framework for the reliable usability evaluation of first-person shooter console video games. The framework includes playthrough evaluation, a structured usability evaluation method adapted from heuristic evaluation.

Usability evaluation can help guide developers by pointing out design issues that cause users problems. However, usability evaluation methods suffer from the evaluator effect, where separate evaluations of the same data do not produce reliably consistent results. This can result in a number of undesirable consequences affecting issues such as:

- Unreliable evaluation: Without reliable results, evaluation reports risk giving incorrect or misleading advice.
- Weak methodological validation: Typically new methods (e.g., new heuristics) are validated against user tests. However, without a reliable means to describe observations, attempts to validate novel methods against user test data will also be affected by weak reliability.

The playthrough evaluation framework addresses these points through a series of studies presenting the need for, and showing the development of the framework, including the following stages,

1. Explication of poor reliability in heuristic evaluation.
2. Development and validation of a reliable user test coding scheme.
3. Derivation of a novel usability evaluation method, playthrough evaluation.
4. Testing the method, quantifying results.

Evaluations were conducted with 22 participants, on 3 first-person shooter action console video games, using two methodologies, heuristic evaluation and the novel playthrough evaluation developed in this thesis. Both methods proved effective, with playthrough evaluation providing more detailed analysis but requiring more time to conduct.

Submitted for the degree of Ph.D.

Department of Informatics, University of Sussex

Monday 10th November, 2014

Contents

Chapter 1

Introduction

1.1 Evaluating Usability

The first barrier that interactive technology must overcome is that of being usable. Poor usability can prevent users from engaging with a product, and for video games in particular, can prevent players from progressing to deeper states of enjoyment and immersion (Brown and Cairns, 2004).

The human-computer interaction community has developed numerous methods to evaluate usability, particularly for traditional systems such as Windows, Icons, Mouse, Pointer based productivity applications. Examples include the less formal "discount" methods such as heuristic evaluation which are quick and easy, and have proven popular both in traditional domains as well as for video game evaluation.

1.1.1 Existing Usability Evaluation Methods Are Unreliable

Despite their common use, usability evaluation methods such as heuristic evaluation tend to produce unreliable results (Hornbæk and Frøkjær, 2008; Jacobsen et al., 1998b; Molich et al., 2004).

In contrast, more structured approaches show evidence of improved reliability, but may not be well suited for game evaluation. This relatively new domain exhibits complexity and non-linear emergence which make it prohibitive to analyse in the level of detail typical of cognitive walkthrough.

This thesis reconciles these two approaches by developing a novel methodology called playthrough evaluation, which combines the design knowledge encapsulated in heuristics, with the reliable evaluation of structured approaches.

1.1.2 The Evaluator Effect Accounts for Some Reliability Problems

The "evaluator effect" is the name given to explain the weak reliability of results produced by different evaluations using the same method and the same data (Jacobsen et al., 1998a,b). The idea is that each evaluator's own subjectivity and personal expertise inform the evaluation, and so each evaluation consequently produces different results. This is problematic as there are

no means to objectively and reliably measure and differentiate the quality of evaluation results.

The evaluator effect is a consequence of the inherent subjectivity involved in making predictions about potential problems that future users might have, and unavoidable interpretations involved in evaluating actual user test data. For example, there are many implicit stages of evaluation that are not explicitly recognised, such as the subjective interpretation of actual observed user behaviour, inferences of the underlying causes, and also predictions of potential future user experiences. What's more, when informal evaluation methods do not clearly define their own operational procedures, evaluators extend their subjective interpretation to how the evaluation itself should be conducted.

Consequently poor reliability introduced at each stage of the evaluation becomes compounded together, and ultimately produces evaluation results which differ substantially even between experts. When this does occur it is typical that differences in interpretations between separate evaluators within the same team will be resolved in private, informal discussions. While this can result in apparent cohesion in the final reports merged from all of the separate evaluators, the specific decisions made during these discussions go unreported. When future evaluations are conducted by independent researchers who were not privy to the previous teams' discussions, it is understandable that the processes and decisions made now will differ to those used by the previous teams. Consequently, each team follows somewhat different procedures and produces different results.

Practitioners may argue that this is a beneficial case of triangulation, where complete agreement is not necessary, and that multiple interpretations of the same issue can in fact be preferable. While this may be true in a idealised sense, it is still important to understand why evaluations differ in order to decide whether these differences do in fact represent constructive triangulation, or are merely aberrations due to errors introduced by the evaluator or deficiencies in the method itself.

1.1.3 Summative Evaluation Requires More Rigour Than Formative

Discount methods such as heuristic evaluation were originally designed to be used throughout the development lifecycle, even during the formative stages of paper prototyping. It is understandably difficult to make reliable and valid predictions about potential future user experience, based only on subjective expert extrapolations of an unfinished product. However, heuristic evaluation can also be used for summative evaluation of a fully functional system. In this condition it should be reasonable to expect reliable and valid evaluation results.

Formative evaluation can introduce additional layers of unreliability:

- It is unclear how valid prototypes are at representing final products. Hence, the set of problems predicted from a prototype and actually encountered with a final product may be substantially different.
- Without actual users to validate against, formative evaluation relies on experts' implicit ability to predict imputed user experiences. This requires a well developed prior understanding of the target users' behaviour and experiences, which is unlikely to be explicitly provided by the usability evaluation method. As such different evaluators inevitably employ different subjective expertise and hence produce different results.

Formative evaluation is most useful to guide the ongoing creative development of a system. The emphasis of game development during the formative stages is to explore core gameplay and the trade-offs between many interrelated, multi-media design elements that continually feedback on each other. Until late in development the core content, mechanics, interface, and controls are usually in a state of creative flux, and so usability is difficult to assess.

Whereas formative evaluation is concerned with potential design, summative evaluation attends to actual implementation, and is most beneficial for evaluating usability with real users and final, working systems. This kind of fully functional system is essential to understand usability and user interaction in first-person shooter games. However, as these systems are usually only available later in the game development lifecycle, summative evaluation approaches are the most appropriate way to address them. Therefore, in order to best attend to the subject this thesis only explores summative usability evaluation.

1.1.4 Structured Approaches Improve Reliability

An increasing number of publications (Baauw et al., 2005; Cockton and Lavery, 1999; A. P. O. S. Vermeeren et al., 2003) suggest that more structured evaluation approaches can ameliorate the evaluator effect.

When informal methods combine each of the implicit stages of evaluation into a single whole it becomes difficult to identify where and why breakdowns in reliability occur. By applying a more formal structure, the evaluation process can be explicitly separated into its component stages, and the evaluator effect can then be examined in more detail in each. For example the effect can be separately observed in each of the following stages, some of which may be glossed over during a discount evaluation:

- Transcription.
- Problem detection.
- Problem categorisation.
- Cause interpretation.
- Outcome evaluation.
- Severity interpretation.
- Future prediction.

By identifying and exposing the particular challenges for reliability in each of the stages it then becomes possible for the methodology to be critiqued and improved more precisely.

1.1.5 The Importance of Researching Usability for Video Games

Usability might not be the key aspect for the success of a game, however, poor usability can act as a barrier to what would otherwise be engaging gameplay. This is especially true for types of games that emphasise the traditional qualities associated with usability, such as efficient information displays and effective input controls. For example, the most critical aspects of

gameplay in first-person shooter games place increased importance on players' ability to quickly and precisely interact, especially in cases of reading and understanding visual feedback, and responding quickly and accurate using input controls. Other qualities associated with playability, particularly as they're found in other styles of games, including character, narrative, and visual aesthetics, tend to be more peripheral in first-person shooter games than the core usability aspects of functional playability.

1.1.5.1 Usability as an Aspect of Playability

There has been a great deal of recent interest in playability and gamer experience. However, no generally accepted, coherent definitions of the terms usability, playability, and gameplay has yet been established in the research community, and indeed some researchers do not expect evaluation standards to emerge any time soon (Ijsselsteijn et al., 2007). Nonetheless, various useful definitions have been developed. For example, Järvinen et al. (2002) identify four aspects that to explain the relationship of usability within playability:

- Functional
- Audiovisual
- Structural
- Social

Functional playability is most similar to conventional usability as it deals with controls, "the input/output element of game-play" and "how well the control peripheral and its configuration is suitable for the requirements of successful gameplay."

Audiovisual playability is also affected by traditional usability concerns regarding the visual usability of the game such as legibility of text on screen or "confusing choices of color". Structural aspects deal more with *gameplay* as they address "the rules, structures and patterns of the product" and involve qualities such as "skill (easy-difficult), experience (enjoyment-frustration), actions (trivial-non-trivial)."

Lastly, social playability is concerned with interactions with other players, as well as issues of society and culture beyond the immediate game itself such as "evaluating what kinds of social practice in media use the product is suitable for."

While playing a game it is first necessary to overcome potential usability problems in the functional and audiovisual aspects of playability. If players have problems understanding displays and executing controls then any evaluation of structural or social aspects may be severely impacted. Once these fundamental issues of usability have been resolved then it is possible to properly assess higher level playability qualities.

1.1.5.2 Adapting Usability Research for Video Games

Games and play in general have been extensively studied in the humanities for a great deal of time (Caillois, 1961; Huizinga, 1955), and likewise visual and narrative aesthetics are well understood by the arts. The main contribution that human-computer interaction has to offer, though, is an extensive and mature body of research on the traditional qualities associated with

usability, such as efficient information displays, and effective input controls. This research was originally developed for other modern computing domains like productivity applications and websites. However, there is a great deal of potential to adapt these practices to other novel areas such as modern video games. First-person shooter games in particular, with their emphasis on functional usability, could potentially benefit the most from this wealth of expertise that the human-computer interaction community has to offer.

With this in mind, the scope of the current thesis is restricted to addressing only those playability aspects that relate to conventional usability. These include some aspects of the Functional and Audiovisual categories, but issues around aesthetic preference and social context are not addressed. Furthermore, in order to maintain a sharp focus the thesis attends specifically to first-person shooter games which have a greater dependence on usability.

Discussion about usability, playability, and games is presented in greater detail in Chapter 2 (Literature Review), Section 2.2 (Usability), and especially Section 2.2.3 (Usability in Game Contexts).

1.2 Research Topic

The research problem is the reliability of summative first-person shooter usability evaluations. This thesis presents the case that weak reliability is a consequence of (1) under-specified methods with a high degree of unreliability being applied to (2) a uniquely complex domain. Discount methods have been adapted for video games, but are too weakly specified to produce reliable results. Traditional structured approaches do not address the unique issues involved in first-person shooter games, and are too cumbersome to apply to such a complex domain.

The purpose of this research then is to reconcile these two different approaches. The result is a novel method, playthrough evaluation, that makes use of the design and evaluation knowledge found in video game heuristic evaluation, but which employs aspects of structured approaches to produce more reliable results.

1.2.1 Research Questions

The starting point of this thesis was to explore usability evaluation methods for first-person shooter games. Following a preliminary literature review in Chapter 2, Chapter 4 (Testing Heuristic Evaluation for Video Games) was devised to attempt a validation of the most promising heuristics in the literature. The study revealed systematic problems of reliability with heuristic evaluation, and so it was decided that the reliability of usability evaluation methods for first-person shooter games would be the core research topic.

The following research questions are addressed by the thesis:

- How reliable is heuristic evaluation for first-person shooter games?
  (Chapter 4, Testing Heuristic Evaluation for Video Games)
- What design and evaluation resources are available in heuristic evaluation, and how can they be represented?
  (Chapter 5, Exploring Evaluation Resource Specificity)

- Can a novel usability evaluation method improve on the reliability of heuristic evaluation when representing these resources in a more structured form?
  (Chapter 7, Testing Playthrough Evaluation)

1.2.2 Reconciling Domain-Specific and Structured Approaches

The central problem of weak evaluation reliability is addressed by more objective, thorough, and rigorous empirical methods. However, it is not conducted uncritically, and an important benefit of this approach is to expose those areas, especially of more aesthetic concerns, where subjective interpretation and opinion is appropriate and necessary, and reliability is unavoidably low. In an interactive entertainment medium that combines function with creativity there must be room for ambiguity, so this thesis identifies its own limits by showing where reliability can and cannot currently be improved. This is an important foundational step, as it provides a sound basis for future research, and signposts the landscape that still needs to be explored in further work.

In addition this approach takes a fine-grained look at the evaluation process and data produced. Rather than using a large number of participants with the hope of generalising relatively high-level, but unreliable conclusions, playthrough evaluation is more concerned with detailed analysis that can be used to validate or support less detailed methods.

Resource-light methods such as heuristic evaluation suffer in that they do not provide sufficient discovery and analysis resources to facilitate highly reliable and valid evaluation (Cockton, Woolrych, Hall, et al., 2003). In contrast, resource-intensive methods such as cognitive walkthrough are impractical for use with complex dynamic domains such as first-person shooter games. The method described here has greater demands for evaluator time and investment than previous discount methods, but pays back this contribution by offering more reliable problem discovery and analysis resources. It provides a relatively heavy-weight method for conducting evaluation, but has perhaps greatest value in validating other methods. Playthrough evaluation can be used to test and support the development of other more light-weight methods, in order to validate their results. There is a clear trade-off involved in using a more complex method. An increased amount of time is required to conduct each evaluation, as more detailed and explicit analysis is involved. However, unlike a traditional expert evaluation in which the expert may have accumulated their expertise over hundreds of hours of industrial practice, the standardised procedures defined by playthrough evaluation can be used reliably by relatively novice evaluators.

1.2.3 Self-Reflexive Methodologies Support Improvement

Furthermore, the research community is best served by exposing and standardising the processes by which evaluation is conducted. By making explicit the decisions that evaluators go through, we as a community are better positioned to analyse and critique them, and as such to improve them. While subjective expert methods have proven highly successful in research and industry, it is difficult and perhaps meaningless to assess the method and evaluator independently. This produces problems when making comparison to other methods, as by necessity the specific evaluators involved in the expert usability evaluation method are themselves impli-

cated in the comparison. What's more, there exists no commonly agreed upon way to test for an expert's ability to conduct an evaluation such as heuristic evaluation. The consequence of this, the lack of definition for what constitutes expertise, and meaningful measures by which to compare between experts, is that each expert will produce different, potentially conflicting evaluation results. With no way to discriminate between more reliable and valid reports, we are left in a position where any one expert's opinion is arguably just as appropriate as another's. Each evaluation will guide product development in a somewhat different direction, and some directions may be better than others. At worse, inadequate "expert" evaluations can provide misleading or incorrect reports, analysis and guidance to developers.

1.3 Contribution

This thesis makes two types of concrete contribution for the human-computer interaction community concerned with first-person shooter usability evaluation:

- Knowledge. More detailed understanding is revealed of the specific causes and consequences of usability problems.
- Methodology. A novel evaluation methodology is presented and shown to improve on the reliability of traditional heuristic evaluation.

1.4 Thesis Overview

The structure and flow of the thesis is described in the following sections, and graphically illustrated in Fig. 1.1 ("Thesis Overview") on the following page.

1.4.1 Chapter 2 (Literature Review)

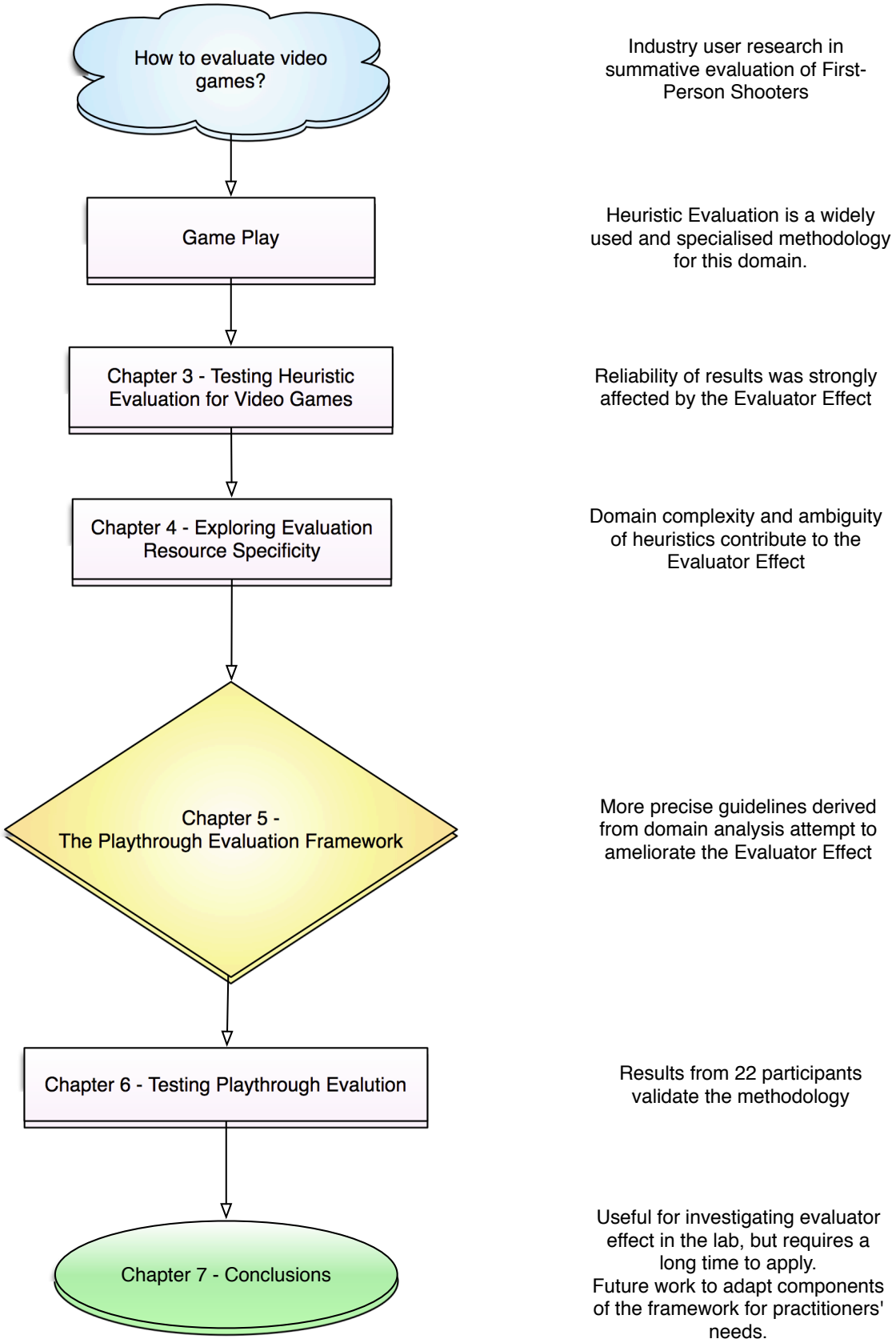
The principal aspects described by the literature review are usability and approaches to usability evaluation methods. In each of these areas, the emphasis is on how applicable the established literature is for use with first-person shooter video games. Strengths and weaknesses are described, and suggestions for improvement or adaptation are proposed. In the latter chapters of this thesis these suggestions are developed into a novel usability evaluation method called playthrough evaluation which ameliorates many of the issues described in this literature review.

The first of the principal topics reviewed is the concept of usability, and how the construct is understood by the research community. In particular the terms effectiveness, efficiency and satisfaction are commonly used, and are employed throughout this thesis. In addition the concept of a usability "problem" is discussed, and several approaches to detecting, describing and classifying problems are introduced.

The second core topic of the literature review is to describe the different usability evaluation methods, including more discount methods such as heuristic evaluation, and more structured approaches such as cognitive walkthrough.

The final topic discussed is the evaluation of usability evaluation methods, both in terms of how different evaluators use the same methodology on the same data, and how different methodologies are compared to one another, with an emphasis on the standard metrics such

Figure 1.1: Thesis Overview



as reliability, validity, and thoroughness. The reliability metric is the main measure investigated in this thesis.

1.4.2 Chapter 3 (Introduction to Studies)

Discusses the background motivation for this thesis, during professional usability work on several well-known first-person shooter games.

Formative development changes rapidly. Game content, mechanics, and controls are usually in a state of creative flux until relatively late in the development lifecycle.

Developers focus on usability last. tutorials and introductions are often not added to the game until the Beta stage, shortly prior to release.

Poetics of First-Person Shooter Games. Functional issues tend to be critical to the player experience, due to the high volume of interaction and precision needed to quickly execute fine-detailed control.

The empirical experience of summative evaluation at Vertical Slice, and the theoretical poetics of first-person shooter games informed the motivation for the studies presented in the following chapters. A series of evaluations were conducted using a variety of approaches: quantitative, qualitative, analytical, and empirical. The novel methodology presented in this thesis, playthrough evaluation, builds on this understanding of mixed methods. In this method evaluators perform both empirical and analytical forms of evaluation.

1.4.3 Chapter 4 (Testing Heuristic Evaluation for Video Games)

The literature review identified heuristic evaluation as the most well developed method for video game usability evaluation, but noted that it in traditional domains it is subject to significant problems of reliability. Therefore, the initial study for this thesis began by exploring the reliability of the method when applied to video game evaluation.

Following Nielsen's approach, 88 issues from real user test sessions were rated against 146 heuristics from 6 of the most promising sets available in the literature, including the main collections designed specifically for video games, and Nielsen's own canonical set. Quantitative analysis revealed systematically poor inter-rater reliability in the ratings of three independent evaluators, and none of the existing heuristic sets were successfully validated.

Similarly to Nielsen's study, principal components analysis was applied to the rating data. Despite the poor inter-rater reliability for individual heuristic ratings, for all three evaluators the principal components analysis produced similar results, and 19 principal components were identified. These 19 components represent underlying areas that groups of similar heuristics address, and represent the high level content of the heuristics.

The chapter concludes that heuristic evaluation for video games is subject to substantial reliability problems due to the subjectivity involved in interpreting ambiguous heuristics. Nonetheless, the 19 components identified are coherent and point to general areas for heuristic evaluation to address. The following chapter examines the heuristics in detail, with the intention of identifying the useful content, and ameliorating the ambiguity that causes weak inter-evaluator reliability.

1.4.4 Chapter 5 (Exploring Evaluation Resource Specificity)

This chapter considers the content and presentation of heuristics, in order to understand whether they could be repurposed in a more reliable form. Specifically the questions asked are,

- What design and evaluation knowledge do heuristics address?
- How is that knowledge represented?
- Can this knowledge be represented in a form that can be used more reliably?

In order to explore why the data from Chapter 4 (Testing Heuristic Evaluation for Video Games) had not produced reliable results, the quantitative data were examined in a closer and more qualitative way. Through interviews with evaluators and inspection of the heuristics and ratings given, it was revealed that evaluators had interpreted the same heuristics in substantially different ways to one another.

The specific presentation of the heuristics was also considered. Through inspection of the heuristic text, a novel observation is made that there are three distinct forms of heuristic with different characteristics that may be amenable to three different forms of evaluation.

The three forms identified are given the names "abstract/reflective"; "analytic"; and "design principles". Each of these is interpreted and used differently in a heuristic evaluation: **abstract/reflective** heuristics such as "Consistency and Standards" (Nielsen, 1994a) are general prompts for an expert evaluator to consider an overall theme or aspect of a system; **analytic** heuristics are phrased with specific, measurable violation criteria which indicate a problem, such as "The player does not lose any hard-won possessions" (Koivisto and Korhonen, 2006); **design principles** are positive guidelines that describe idealised examples, but which may not in fact be necessary or even appropriate in all cases. For example "The game uses humor well" (Desurvire and Wiberg, 2009).

Without clear guidance regarding how to evaluate the individual criteria for overall violation or conformity of the heuristic as a whole, evaluators used their own subjective opinions and arrived at incoherent evaluations.

In order to address these problems, the chapter proposes an approach to decomposing heuristics into more specific and measurable criteria, similar to the way in which scenario-based design Claims Analysis examines scenarios in order to produce usability claims for testing.

1.4.5 Chapter 6 (The Playthrough Evaluation Framework)

Having identified the potential to restructure the design and evaluation knowledge of heuristics into a more concrete and specific form, this chapter systematically applies the approach and presents the player action framework and playthrough evaluation methodology.

Heuristics from each of the 19 high level areas were examined, and specific, measurable criteria that are implicated in each heuristic were identified. Issues from Chapter 4 (Testing Heuristic Evaluation for Video Games) were similarly examined, and represented using the criteria referred to by the heuristics. A taxonomy of interaction events, breakdowns, outcomes and game components was thus produced, using a common terminology shared between issues

and heuristics. Having identified the specific content relevant to an evaluation, the next phase was to represent that content in a form that can be used in a more reliable manner.

Chapter 2 (Literature Review) showed that more structured approaches can improve on the reliability of usability studies. The content identified was therefore restructured into a novel hierarchical tree form, called the player action framework. At the top level are the 19 areas identified by the principal components analysis. Below this are specific criteria for evaluation, derived from the heuristic and issue analysis.

This structure separates the underlying cause of a usability breakdown from the resulting outcome, which provides extra analysis and discovery resources for evaluators to use in order to determine what has occurred, why it happened, and what needs to be resolved in order to fix the problem.

The high level areas are used by evaluators as prompts to reflect on for potential usability issues. Once a candidate issue is identified, the second level provides a coding scheme of events and criteria to document the actual or potential interaction patterns involved. This lowest level assists the evaluator in identifying the underlying design issues that have caused the interaction patterns.

1.4.6 Chapter 7 (Testing Playthrough Evaluation)

This chapter empirically tests playthrough evaluation, and shows a comparison against conventional heuristic evaluation. Playthrough evaluation requires the evaluator to play the game as a normal participant would, but additionally to review and code their own footage after the play session, using the transcription coding scheme described in the preceding chapter. Playthrough evaluation and heuristic evaluation were conducted on the same game and the standard metrics of reliability for problem identification and classification were computed. Playthrough evaluation is shown to produce a detailed analysis, with thoroughness and reliability comparable to the data reported for other usability evaluation methods in the literature.

1.4.7 Chapter 8 (Conclusions)

The final chapter of the thesis concludes by reiterating the initial research questions and reflecting on how well the novel playthrough evaluation framework has addressed them. The novelty and contribution of the thesis are considered and a case presented for the contributions made to the research community. Limitations of the approach are discussed, including issues around how generalisable the method is with regards to the demographic of players and evaluators. Playthrough evaluation is also open to adaptation for other genres of game. What's more, further work is indicated to expand the methodology to address issues beyond usability, moving to more complex questions of playability. Lastly, additional studies are proposed to explore the representative validity of prototype systems in order to extend the work to formative evaluation.

Chapter 2

Literature Review

2.1 Introduction

This chapter gives an overview of the literature on usability evaluation used in traditional domains and adapted for video games. Many of the key concepts and terms used in latter parts of the thesis are introduced, including an important core concept, the evaluator effect, which becomes a central concern when considering each of the methods in the literature. Particular attention is given to different usability evaluation methods, showing their strengths and weaknesses, and discussing their applicability to video game evaluation, especially heuristic evaluation as the most widely used method for evaluating games. However, evaluation for video games is shown to be more complex, and less reliable and valid than traditional domains. Following the review, the subsequent thesis chapters develop and test methodological improvements to resolve the issues identified.

2.1.1 Research Questions

The following specific research questions are posed for this chapter,

- What is usability, usability evaluation, and how does it relate to product development?
  Section 2.2 (Usability)
  Section 2.3 (Evaluation)
- What are the problems with evaluation methods documented by the literature?
  Section 2.4 (Evaluator Effect)
- What are the criteria and procedures for usability evaluation?
  Section 2.5 (Metrics)
  Section 2.6 (Heuristic Evaluation)

2.1.2 Overview of Literature

The literature review is divided thematically. The main concerns of usability and the evaluator effect are briefly introduced in the following sections, explaining why they are reviewed, and what contribution they make for the thesis as a whole.

2.1.2.1 Usability Supports Game Play

Usability is a foundational aspect of user experience, and especially aspects such as controls and feedback have been shown to prevent more refined player experience states such as immersion in video games (Brown and Cairns, 2004). Prior to addressing these more insubstantial states it is first necessary to develop a solid foundation on the more concrete issue of usability for games. This critically important first stage has been underdeveloped in the literature to date, so is the core topic of this thesis. The scope of research is further constrained to only consider the process of game play, and not to address broader issues of context, motivation, media effects, or cultural significance. Future work will expand from usability to develop evaluation methodologies for more nebulous areas such as playability and player experience.

Usability in traditional domains is usually concerned with qualities such as Effectiveness, Efficiency and Satisfaction (ISO, 1998). While traditional applications want to help the user achieve their productivity goals as effectively and efficiently as possible, the real *goal* of a video game is arguably to give the player an entertaining experience. This experience can include a subtle blend of frustration, efficiency, failure, and effectiveness, which in turn creates a complex form of *player experience "satisfaction"*. This is not the same as user experience satisfaction based on the more simple usability qualities. Whereas traditional domains should try to lead the user through to their goals as effectively and efficiently as possible, a counter example is given of a theoretical video game with just one clearly visible, accessible, and *usable* button: "press to win". This would certainly not meet the needs of a satisfying gaming experience. However, caution should be taken not to fall into the extremes. While such an example of total usability would make for a very poor gaming experience, the opposite is also true: a video game that completely ignores usability principles will be unplayable.

In particular, it is still necessary to be able to detect and measure such occurrences in order to carefully manage the experience. The domain of video games, and specifically first-person shooter games, are more complex and demanding environments to evaluate, even for the more straightforward concerns of usability. This is the principle topic of the thesis.

2.1.2.2 Evaluator Effect

A core interest of this thesis which is explored in great detail in this literature review is the evaluator effect.

The evaluator effect is defined as the discrepancies between evaluators results. This chapter shows that the effect occurs across a number of different evaluation methodologies, and throughout their various stages of use. Subsequent chapters develop an approach to ameliorate and control for this effect. The literature suggests that the evaluator effect can be managed through employing more detailed, structured, and analytic approaches to usability evaluation (Cockton and Lavery, 1999).

The next section shows how the effect can occur in numerous stages of an evaluation. By breaking down the procedures and examining them in more detail with these two effects, we are in a stronger position to be able to identify where sources of low reliability come from. This in turn allows us to better test and improve our methodologies.

2.1.2.3 The Evaluator Effect Throughout the Evaluation Process

Evaluation consists of the following abstract stages, each of which have the potential to contribute to the evaluator effect,

- Problem discovery.
- Problem analysis.
  Cause evaluation.
  Outcome evaluation.
- Problem Matching.
  Duplicate filtering.
  Grouping / categorising / merging.

Errors can be introduced at each stage, which compound the reliability problems in each subsequent stage. For example, if evaluators disagree at the problem discovery stage then all of the subsequent stages will exhibit reduced reliability. i.e., if one evaluator discovers a particular problem, but the other evaluator does not, then any conclusions based on summaries or computations of mean averages of problem existence, cause, outcome, duplicates, or categories will necessarily be different as well.

A brief introduction to each stage is presented in the following, with more detail in later sections of this literature review.

Problem Discovery

The evaluator effect initially occurs when different evaluators notice different things as being candidate problems. These are reported as individual "problem tokens".

The process of problem discovery can be broken down into sub-stages,

- Noticing an issue.
- Considering the issue problematic.

In most user test and heuristic evaluations where formal procedures are typically not defined for how to detect problems, evaluators proceed in a more-or-less informal, ad hoc, free form manner. Most methodologies do not specify how evaluators should detect problems, and so leave it up to interpretation rather than a more objective, systematic approach. A consequence of this is likely to be low inter-evaluator reliability for problem detection rates. By defining more rigorous, repeatable and measurable procedures it should be possible to make improvements in this area.

During problem detection, evaluators make an implicit judgement about whether a particular candidate issue is problematic or not. This usually occurs prior to any formal analysis of the candidate itself. There is an inevitable tradeoff between the speed of making an informal decision and the increased reliability and validity that could be produced if a more formal, systematic approach were used to analyse all possible candidates. Highly structured methods, such as cognitive walkthrough take the latter approach by defining the correct sequence of events in the most possible detail. For first-person shooter evaluation a more balanced approach would be preferable.

Problem Analysis

Analysis of issues is also often conducted in an informal way. This lack of process leads to further reliability problems. Even in cases where evaluators are presented with a fixed set of pre-determined issues, if they are not also provide with a reliable framework for analysing them evaluators are likely to disagree in their interpretations.

Analysis of an issue consists of two main parts: analysis of the design features and other factors that caused the problem, and analysis of the resultant outcomes. In the case of an observational user test, the outcomes should be more reliable to document when all evaluators use the same data - that is, the observations of the user. However, problems introduced in the previous section, problem detection, may mean that different evaluators do no address the same issues, even when observing the same user test session. Analysis of the factors that contributed to the problem is even more difficult than analysing outcomes, and so reliability for this is expected to be lower.

In some cases analysis is more or less a question of simple categorisation. One example is heuristic evaluation, where typically each issue is assigned a single heuristic that best describes the issue. However, this can be a source of disagreement as typically heuristics do not separate problem cause and outcome, so evaluators can in effect be evaluating different aspects of the same issues. Heuristics instead are a single, compound construct that attempts to address all aspects of a problem. This introduces a wide range of possibilities for evaluators to disagree on the appropriate categorisation to represent the issue.

These differences are again even more significant in the case of evaluators making predictions about potential user experience, rather than interpreting actual user test data. As predictions are purely based on individual experts' opinions, results are likely to differ by expert and so show low reliability, hence validity will vary by evaluator too.

The remaining degree of inter-evaluator disagreement produced even when following a more formal procedure is appropriately termed evaluator effect. This may be due to the evaluator's inability to follow the procedure correctly (which could be tested, so facilitating training and evaluation of evaluators), or may be due to unavoidable interpretation involved in the method.

In terms of identifying and describing outcomes, it should be possible for observers of user test sessions to produce reliable results. Observing and documenting performance type usability measurements in particular should be relatively straightforward. For example counting error rates, and measuring time-on-task, etc. How feasible this is, however, may need to be considered further. This is particularly relevant in a fast-paced first-person shooter game, where players are intentionally challenged, play is dynamic and emergent, and players are expected to experience a certain degree of failure. Compared to more simple, and relatively static traditional interfaces, it may be less clear what constitutes an acceptable or unacceptable error in a game.

Related to the issue of problem analysis is the issue of how problems are reported. In many cases, such as heuristic evaluation, reports are usually made as free form text descriptions of problems, without interactive multimedia data such an example of the design feature or footage of an actual user test incident. This separation of original source material and resulting summarised analysis and report can introduce further errors in subsequent stages of the

evaluation process. In particular, during the next analysis stage, problem matching, analysts generally have to interpret these secondary sources of evidence rather than the original source material.

Problem Matching

Once individual problem tokens have been analysed the final stage of an evaluation is typically to summarise the results. This involves aggregating all of the reported problems, noting duplicates and defining unique problem types. This may also include a frequency count, showing how many individual problem tokens were reported for each general problem type.

However, errors introduced in any of the preceding sections will affect the results of this final stage.

This aspect of the evaluator effect has been called the matcher effect in the literature (Hornbæk and Frøkjær, 2008). This describes the difference in the way separate analysts merge and distinguish problem tokens. Given a set of problem tokens, different analysts may disagree over which of the tokens are the same as one another, and which are different. This is especially likely with the kind of informal practices typical of heuristic evaluation, for example. More clearly defined methods could help guide the process, and more structured approaches to analysis and reporting could help to produce data that can be more objectively treated.

E. L.-C. Law and Hvannberg (2008) describe two distinct stages involved in usability problem (UP) comparisons,

> "...*filtering*, that is, to eliminate duplicates within a list of UPs identified by a user when performing a certain task with the system under scrutiny or by an analyst when inspecting it"
>
> "...*merging*, that is, to combine UPs between different lists identified by multiple users/analysts, to retain unique, relevant ones, and to discard unique, irrelevant ones."

They comment on how little is known or reported about this stage of the evaluation process,

> "...in the HCI literature, the UP consolidation procedure is mostly described at a coarse-grained level."
>
> "...the actual practice of UP consolidation is largely open, unstructured and unchecked."

Their study concludes that when teams of evaluators work together to merge problems, *the number of UPs reported is deflated*, and *the frequency and severity of certain UPs inflated excessively*. This has clear implications if these results were to be acted on by developers. The inflation of certain problems, and the deflation of others, undermines the validity of the results, and may lead developers into incorrect or poorly prioritised redesigns.

Observation is Less Affected Than Prediction

These stages of evaluation are the same for observation studies, such as in user tests, as well as for prediction studies, for example in expert inspection such as heuristic evaluation. In both cases reliability is a concern, but additionally in the case of a study that makes predictions, validity is also involved. The addition of this extra factor increases the potential evaluator

effect. In cases where expert opinion is employed rather than the analysis of representative, empirical user test, the evaluator effect is expected to be greater. This predictive interpretation is compounded with problem categorisation, such that the expert's evaluation predicts not only *that* problems will occur, but that *certain types* of problems are likely to occur if design changes are not made. Rather than relying on an unfinished prototype, predictions for possible problems could be based on a stable, finished product. This allows the possibility to test the validity of the expert's predictions by comparing them to the actual experience of real users with the real product the predictions relate to.

2.2 Usability

2.2.1 Introduction

Usability as a concept is multi-faceted, with a diverse array of definitions. This section reviews the literature, and shows how these diverse definitions lead to a multitude of ways to evaluate it. The conclusion provides a definition of video game usability for the scope of evaluation in this thesis.

2.2.1.1 Research Questions

The following research questions guide this literature review, with examples of the sections that address each shown below,

- How is usability defined?
  Section 2.2.2 (Concept).
- What role does usability have in video games?
  Section 2.2.3 (Usability in Game Contexts).
- When is usability evaluated?
  Section 2.2.4 (Usability in the Product Lifecycle)
- What is a usability problem?
  Section 2.2.5 (Usability Problem).
- How is usability evaluated?
  Section 2.3 (Evaluation).
  Section 2.6 (Heuristic Evaluation).
- What are the challenges to usability evaluation?
  Section 2.4 (Evaluator Effect).
- How is usability evaluation measured?
  Section 2.5 (Metrics).

2.2.2 Concept

Traditional Definitions

The ISO standard for usability is specified in terms of *Effectiveness*, *Efficiency* and *Satisfaction* outcomes,

> "The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use."
> ISO (1998)

In turn, the following formal definitions are given,

Effectiveness:
"...accuracy and completeness with which users achieve specified goals."

Efficiency:
"...resources expended in relation to the accuracy and completeness with which users achieve goals."

Satisfaction:
"...freedom from discomfort, and positive attitudes towards the use of the product."

Usability is Objective and Subjective

In addition to acknowledging these core aspects, Ham (2008) notes that usability can further be understood through two paradigmatically different lenses: objective and subjective, where the objective components address the assessment of task performance, and subjective components attempt to address how the users feel about the usability of the system.

Cockton (2012) introduces and problematises the concept of usability, showing how it has developed historically, with particular attention to formal standards and seminal papers in the literature (Gray and Salzman, 1998; Hertzum and Jacobsen, 2001; ISO, 1998, 2001; ISO/IEC, 2005). A shift is described from an system-centric, homogenous, "essentialist" model of usability, towards a more interaction-centric, heterogeneous, "contextual" model of quality-in-use and user experience. Improving the usability of a system is then understood as decreasing the *cost* of usage, but does not necessarily improve the *value* of the user experience.

Usability in Development and Use Context

Cockton (2012) considers the concept of usability, and examines some fundamental propositions about it and its development in human-computer interaction and interaction design, with attention paid to a number of seminal texts charting crises in the research community and changes in formal definitions, including: Gray and Salzman (1998); Hertzum and Jacobsen (2001); ISO (1998, 2001); ISO/IEC (2005).

The paper discusses two different positions implicit in the literature regarding where and what usability is:

- An inherent property of a system (the "essentialist" rhetoric), where all of the causes of user performance come from the technology of the system, and which can therefore be examined with system-centred methods.
- The result of the interaction between users and a system within a situated context ("relational" or "contextual"), thus requiring a nexus of several methods to identify causes of issues.

There is a balance between these two positions, though the contemporary definitions (ISO/IEC, 2005) tend to emphasise situated quality-in-use. Usability evaluation then takes place within a development process for a particular design agenda. It is noteworthy that the chapter asserts that neither the essentialist nor contextual approaches define what constitutes evidence of positive or negative usability. This question of evidence becomes an important consideration for the novel methodology presented in Chapter 6 (The Playthrough Evaluation Framework).

2.2.3 Usability in Game Contexts

Shackel (2009) gives an initial definition of usability in rather conventional terms,

> "...the capability in human functional terms to be used easily and effectively by the specified range of users, given specified training and user support, to fulfil the specified range of tasks, within the specified range of environmental scenarios."
>
> "...the capability to be used by humans easily and effectively"
>
> "Easily = to a specified level of subjective assessment."
>
> "Effectively = to a specified level of (human) performance."

However, the definitions become more interesting when they are fleshed out with proposed numeric metrics across four main dimensions,

- Effectiveness
  - Time to complete task.
  - Number of errors during task.
- Learnability
  - Time to complete training.
  - Amount of training and support.
  - Time to relearn.
- Flexibility
  - Percentage variation in tasks.
- Attitude
  - Acceptable levels of tiredness, discomfort, frustration, and personal effort.
  - Satisfaction causes continued and enhanced usage.

While the first two dimensions are relatively typical for general definitions of usability, the last two items are particularly interesting from the perspective of game usability. First, flexibility explicitly acknowledges that there may be acceptable degrees of variation in task behaviour. This is a feature that is often overlooked for more simple domains, or in usability evaluation methods that define specific sequences of interaction such as can be the case with cognitive walkthrough. This is especially pertinent in the context of a first-person shooter video game where much of the interaction is emergent and impractical to define at that level of detail. Additionally, this definition of Attitude provides a more nuanced and appropriate understanding of the player experience of usability than provided by traditional notions of Satisfaction. Specifically it implicitly acknowledges that some degree of negative affect could be acceptable, as long as the user is overall satisfied enough to keep using the system. In a video game that is intentionally challenging, and where struggle and failure are to be expected, these are important considerations to take into account.

Of additional interest to this section is part of the Cockton and Woolrych (2009) series which proposed 10 core usability terms, particularly because the team defining these terms included a professional game designer:

1. Learnability.
2. Responsiveness.
3. Adaptability.
4. Trustworthiness.
5. Accessibility.
6. Excitement.
7. Challenge.
8. Efficiency.
9. Satisfaction.
10. Complexity.

The term "complexity" is explored in further detail, and its definitions are of particular interest to this thesis. In their study, 3 definitions for complexity emerged. The first applies to all system types and deals with difficulty in task performance, such as where the number of task steps is excessive, or where task steps are not well tied to task-model elements. The second applies to process control systems, and is concerned with the level of detail in the user interface and underlying process. The third definition for complexity is explicitly defined for game systems. It deals with increasing levels of challenge for engaging and entertaining players. Characteristics of problems for this type of challenge include: user dissatisfaction; over-competent performance; and inability to reach a competitive level of performance. Note that many of the heuristics in the games literature are concerned with these issues, as well as those from the first category,

2.2.3.1 Playability / Game Usability

There exists no cohesive, validated, and accepted standard for the term "playability" in the literature. The relationships between usability, playability, fun, and games are worth mapping out in order to understand how the research community defines these terms, and to specify how they will be used in this thesis. This section outlines the diverse definitions, and concludes by outlining how the term is used in the studies. In particular, Chapter 5 (Exploring Evaluation Resource Specificity) shows the results of a statistical analysis of heuristic evaluation, and suggests a taxonomy of heuristics to address game usability for the scope of this thesis.

The definition of a general model of playability, crossing multiple genres, remains an interesting research project for future work.

First-Person Shooter Gameplay

There is a great deal of diversity in the types of interfaces, interaction, and gameplay experiences across the spectrum of video games. Each different style may be more amenable to different kinds of evaluation, so in order to maintain a tight focus this thesis only attends to first-person shooter games. This type of game is particularly interesting as a research subject

due to its substantial differences from traditional subjects of usability evaluation. More simple games for children are already well served by existing usability evaluation methods, particularly those that involve very detailed examination of the player-system interaction (Barendregt et al., 2003; Barr, 2010b). While there is no single accepted formal definition for first-person shooter games, typical characteristics include fast-paced gameplay, a first-person perspective, and an emphasis on embodied action rather than narrative.

This has implications for the working definitions of usability and playability. In particular, the emphasis will be placed on the most critical aspects of gameplay in this style of game, which include functional aspects of usability including motility and controls. In contrast, the reduced impact of narrative in this style of game mean that the definitions used will attend to those aspects much less.

The methods and results presented here may apply to games which are usually identified as belonging to other genres, and perhaps may not apply to all games which would normally be considered to be first-person shooter. For example, a game could present a first-person perspective, and use shooting as a core mechanic, but feature rather more RPG characteristics. Likewise, other games which do not involve the first person perspective could still benefit from the methodology presented here.

HCI and Games

Bernhaupt et al. (2008) comment that usability evaluation methods derived from productivity software tend to be used during game development and form the basis for much video games research in human-computer interaction, but that no coherent, comprehensive, formal framework exists. While the relationship between human-computer interaction and games is not straightforward, there are areas of productive research that need to be explored further, such as usability and evaluation. As both human-computer interaction and video games are relatively new areas of academic research, it is only recently that the two have begun to develop and find parallels. Bernhaupt et al. (2007) raise the question of how to evaluate usability and user experience in games, and specifically address the question of what human-computer interaction can do to assist the study of video games.

Jörgensen's seminal review of human-computer interaction, usability and games (Jørgensen, 2004) lists a number of common themes in the two disciplines: learning; motivation; mental models; control; interaction; feedback; spatial navigation; linguistic and visual expressions, etc. These seem to suggest an initial set of areas for a definition of game usability. He is keen to point out that a significant difference is with the role of challenge, which he contrasts with the role of utility in productivity software. While games are often described as being "easy to learn, but difficult to master," conventional usability would typically prefer "easy to learn and easy to master."

Barr considers video games to be only superficially within the realm of traditional human-computer interaction, citing their intentional difficulty, imposed constraints and predefined tasks as being examples of the differences between them and productivity applications (Barr, 2008; Barr et al., 2006, 2007). Juul and Norton (2009) also suggest that games intentionally require players to take less efficient means of achieving an objective, and that effort must be

taken to design inefficiency into the player's tools. This is framed in opposition to conventional human-computer interaction where the assumption is that computer systems are inherently difficult and effort must be taken to smooth the interaction between human and computer. Barr also observes that while video games are indeed software, they are also *games* and an entertainment medium, which have a unique quality that human-computer interaction does not traditionally address: gameplay as a process, rather than the results of processes, as human-computer interaction would typically attend to in terms of efficiency or productivity. Juul similarly concludes that games can be characterised by a diversity of both easy and complex interfaces and gameplay precisely because they must be seen not only as software that humans interact with for a goal, but are fundamentally an emotive medium and an activity in which the beauty is in the doing. In an early paper on the subject, Carroll and Thomas (1988) attempt to unpack the distinction between fun and easy, citing games as a case in point. Monk et al. (2002) note that traditional human-computer interaction with its emphasis on objective metrics of usability has not addressed subjective and hedonic experiences of fun. Schaffer (2009) identified usability issues as potential barriers to enjoyment, or as a necessary but insufficient condition for a good player experience.

2.2.3.2 Play and Fun

Many researchers have attempted to address the apparently simple and straightforward notions of "play" and "fun", though there are still no formal definitions which are widely accepted by the community.

Motivation is clearly an important component that is addressed by much literature in psychology as well as game studies (Bartle, 1996; Järvinen, 2008; Johnson and Gardner, 2010; Kellar et al., 2005; Komulainen et al., 2008; Meurs, 2007; Rigby and Ryan, 2007a,b; Ryan et al., 2006; Schuurman et al., 2008; Sherry and Lucas, 2003; Tychsen et al., 2008; Vorderer et al., 2003; Ward, 2010; Whitton, 2007; Yee, 2006a,b; Zammitto, 2010). Indeed, one of the earliest papers on games and heuristics was the seminal Malone (1980) [1]. This publication principally looked at intrinsic motivation for educational games. The potential research provided by the gaming community has not gone unnoticed by human-computer interaction researchers, and efforts are underway to distill game designers' expertise into a format that can be leveraged in traditional domains (Hassenzahl et al., 2008; Pausch et al., 1994).

In most cases, usability is defined in line with Herzberg's Two-Component Theory (Herzberg, 1973). This theory addresses motivation at work, and defines two opposite components: positive motivators; and negative hygiene components. In addition to these positive and negative qualities the theory defines absence of a component as neutral valance, such that, for example, the (neutral) absence of a (positive) motivation components does not produce (negative) dissatisfaction. Likewise, the (neutral) absence of (negative) hygiene components does not produce (positive) motivation.

Usability is usually focussed solely on negative hygiene components, or usability *problems*. In contrast, user experience is much more concerned with both positive and negative components (Cockton, 2012).

[1] See also Malone and Lepper (1987)

2.2.3.3 Gameplay

The term gameplay usually refers to qualities such as challenge, game mechanics, and non-player character behaviour. It is usually used as if implying a noun, such as "The gameplay in *Half Life* is very good". This is similar to the way that the term "usability" sometimes implies a system-centred, "essentialist" characteristic of a product (Cockton, 2012). The broader term "game play" as a verb means the process or act of playing a game.

In their seminal paper, Robin et al. (2004) proposed the distinction between game mechanics, dynamics, and aesthetics. Mechanics are the components of the game, actions, behaviours, and controls that the player can use. Dynamics describes the the emergent behaviour of the mechanics, the dynamic result of player's use of those game components and controls. Mechanics and dynamics work together to create game aesthetics, the player's emotional response to the game.

Giddings (2006) presents a thorough analysis of the terms, creating a novel formulation of "game/play/er". This visually indicates the inter-relation between the game, the player, and the play that occurs in between the two agents when they come together. In this definition, gameplay is understood as a temporal and inter-relational *event* rather than an intrinsic quality.

Often the terms are not clearly defined, but it is possible to infer categories of meaning by a close reading of their use. For example, Federoff (2002) uses "game play" to refer to challenge,

> "Game play is the process by which a player reaches the goal of the game."
> "Game play includes the problems and challenges a player must face to try to win the game."

Heuristics that refer to gameplay include,

> "Game play should be balanced so that there is no definite way to win."

In contrast, Desurvire and Wiberg (2009), uses "game play" mainly to refer to the overall experience, including qualities such as "Variety of Players and Game Styles" and "Enduring play".

In contrast to "Game Play" heuristics, the other high-level categories include:

- Coolness/Entertainment/Humor/Emotional Immersion.
- Usability & Game Mechanics.

The MIPA framework (Lee and Im, 2009) provides general usability heuristics as well as game-specific qualities, which are addressed across four divisions,

- Game mechanism.
- Game interface.
- Game play.
- Game aesthetics.

Malone's pioneering set of heuristics (Malone, 1980) was arbitrarily categorised into three sections: Fantasy, Challenge and Curiosity. The challenge category is most closely aligned with usability and playability, with the other two addressing more aesthetic concerns.

Heuristics as Usability and Playability Indicators

Desurvire et al. (2004) defines four categories of heuristics which also make some distinction between usability aspects and game or play aspects,

1. Game play
   "...set of problems and challenges a user must face to win a game"
2. Game story
   "...plot and character development"
3. Game mechanics
   "...programming that provides the structure by which units interact with the environment"
4. Game usability
   "...interface and ... elements the user utilizes to interact with the game (e.g. mouse, keyboard, controller, game shell, heads-up display)."

The game usability category emphasises the interface components of the system, whereas the game play category treats the goals and use of the system as separate components of the game. Similarly Omar and Jaafar (2009) distinguish gameplay as a discrete subject of evaluation. They categorise a number of heuristics according to the following top-level groups:

- Interface.
- Multimedia.
- Content.
- Educational / Pedagogical.
- Playability.

Most of these heuristics deal with traditional usability of different system-centric types, in addition to the educational purpose of that particular study. The heuristics in the Playability group primarily address different aspects of challenge, but also incorporate other forms of traditional usability:

- Challenges provided are up to the users standard/level.
- Users able to strategise.
- The pace of the game is in balance.
- Players able to control the game.
- Progress of the game can be seen at anytime.
- Players able to perform to their best ability.
- Challenge is adequate - not too easy and not too difficult.

These distinctions begin to suggest criteria that can be used to evaluate the different aspects of usability and gameplay.

Febretti and Garzotto (2009) conducted a meta-review of the heuristics in the literature, and informally and subjectively selected 22 that they felt dealt with playability, and categorised them into 7 groups:

1. Concentration and Immersion
2. Challenge
3. Player Ability
4. Control
5. Objectives and Feedback
6. Social Interaction
7. Artificial Intelligence

These groups mix elements of the system with qualities of player experience. In addition, 15 usability heuristics were derived from the literature, making some adjustments where the authors felt they were needed, and categorised into 5 groups,

1. Customization
2. Controls
3. Game Views
4. Interface/Layout
5. Game Menu

These categories refer more to components of the game, but are defined in terms of the player's use, such as prescribing that the interface "should be intuitive and immediate to let the player keep focus on gameplay".

Usability of Game System Components

Laitinen (2008) describes the components of the game system that are the subject of usability evaluation:

- Screens
- Menus
- Displays
- Controls
- "other possible user interface elements that the player uses before, during and after playing the game."

This inventory considers usability insomuch as it is an artefact of the system. This corresponds to the "essentialist" notion of usability in the terminology from Cockton (2012). In fact, it goes further by stating that the goal of evaluation is to ensure that the *user interface*:

- "is easy to learn"
- "is fluent to use"
- "supports interactions typical for the game."

Playability of System and Player Components

Nacke (2009) presents a brief and high-level suggestion for a theoretical, hierarchical game usability model. He proposes 3 separate areas of research with their own methods and interests:

- Technology is to be measured using standard Quality Assurance techniques.
- Player experience or user experience asses both objective and subjective gameplay.
- Social and Community measures can be conducted with anthropological or sociological studies.

This separation of the different aspects of usability into system, player/user, and context is also seen in other literature. For example, Yue and Zin (2009) conducted a meta-review of the literature and proposed a taxonomy of 6 constructs for usability evaluation of pedagogical games:

1. Interface
2. Mechanics
3. Gameplay
4. Playability
5. Feedback
6. Immersion

Järvinen et al. (2002) similarly begins to make a taxonomy of system and player criteria that can be used to evaluate playability,

> "'playability' is developed here to function as a similar evaluation tool and research discipline as usability. Playability is, in this sense, a collection of criteria with which to evaluate a product's gameplay or interaction."

He goes on to offer a clarification between four different aspects of playability:

- Functional
- Structural.
- Audiovisual
- Social

Functional playability is most similar to conventional usability, as it deals with control mechanisms, peripherals and player's ability to use the game, "the input/output element of gameplay." In regards to the specific context of gaming, functional playability is concerned with "...how well the control peripheral and its configuration is suitable for the requirements of successful gameplay." Structural playability is related, addressing how difficult and enjoyable the structural rules and game mechanics are, "the aesthetics of digital games and entertainment." Audiovisual playability deals in part with the visual usability of the game (e.g., legibility of text

on screen,) but also more aesthetic questions of style and preference. Social playability is concerned with the context of use, and "what kinds of social practice in media use the product is suitable for".

Several suggestions of criteria are provided for how these four elements should be evaluated, though no specific formal method is defined, and the criteria themselves are not elucidated with reference to metrics that would facilitate reliability testing,

Functional.
> "...participants will be questioned on the axis intuitive-non-intuitive and on their experiences on the orthogonality/contextuality of the controls."

Structural.
> "The study of formal aspects will be an expert evaluation where the rules, structures and patterns of the product will be explained."
> "...play-testers will be asked to give their evaluation on the following qualities and axis: skill (easy-difficult), experience (enjoyment-frustration), actions (trivial-non-trivial)."

Audio-visual.
> "...the evaluation axis runs from photorealism to caricaturism and abstractionism"
> "...the combination of so-called dimensionality (2D, 3D, isometric) and point of perception (1st or 3rd person) of the product is evaluated in light of the product genre and rules (in the case of a game)."
> "...detailed observations on possible problems, such as confusing choices of color, and the possible inconsistencies of the game world."
> "...audiovisual qualities will be evaluated based on its relation to other similar products."

Social.
> "...evaluating what kinds of social practice in media use the product is suitable for, i.e., whether the functional playability is suitable for adaptation to platforms such as digital television or mobile phones with restricted input devices, and on a general level, what kind of digital entertainment is suitable for different contexts of use."

The scope of the current thesis is restricted to addressing only those playability aspects dealing with conventional usability issues. These include some aspects of the Functional, Structural and Audiovisual categories, but issues around aesthetic preference and social context are not addressed.

Evaluating Gameplay as Distinct to Usability

In contrast to usability evaluation, when evaluating gameplay, Laitinen (2008) makes it clear that the subject under consideration is not the interface *per se*, but rather the mechanics and interactions that occur. The goal of gameplay evaluation is to,

- "...find and remove the challenges that are not intended by the game developers"
- "...make sure that the gameplay is as fluent as fun as possible."

Examples of gameplay problems further elucidate Laitinen's differentiation from usability:

- Boring and repetitive tasks.

- Unclear goals.
- Unfair punishment for failure.

In each case, these criteria are defined in subjective, user-centric ways. This is in contrast to the more system-centric definition of usability. Defining terms like this with even such subjective criteria still suggests some ways to evaluate usability.

What is a Problem in Game Usability?

In a study of direct manipulation devices, such as a word processor, Springett (1998) discusses what constitutes a genuine error. The case is made that often if a user makes a mistake but is able to rapidly recover and then succeed with their task then this should be classified as a "non-error" in the genotype of his study. Incidents in this category are not considered to necessarily require a design change, as a certain amount of experimentation and error is expected from the user.

At face value this sounds like an appropriate assessment for games too. However, the position argued by the present thesis is that it is first necessary for multiple evaluators to reliably document what has occurred in terms of user interaction. Following a suitable level of agreement of observed events, an informed decision can be made as to the impact of the events, and hence agreement regarding whether these events constitute the need for a design change or not. The general approach advocated here and made explicit in later chapters is to expose and make explicit the evaluators' decision making processes. This facilitates a more detailed analysis of reliability and the evaluator effect.

Usability as Barrier to Gameplay

Fabricatore (1999) provides a definition of playability as,

> "how well the player can understand and control the fundamental elements of the game-play (i.e., the way she can understand what can be done, what must be done, and to actually do it)"

This is developed in the later Fabricatore et al. (2002),

> "Playability is the instantiation of the general concept of usability when applied to videogames, and it is determined by the possibility of understanding or controlling the gameplay."

They present a hierarchic model of playability derived through a grounded theoretical analysis of qualitative player reports. Design prescriptions and recommendations at each node express heuristics for design and evaluation.

Schaffer (2009) expresses an understanding of usability defined in relation to the Four Fun Keys from Lazzaro (2008). Here, the potential for enjoyment expressed in each of the four keys is modulated by the usability of the game, resulting in Overall Enjoyment. Game usability, though, is only considered to be a feature of the game. In the terminology of Cockton (2012) this is a heterogeneous, system-centric, *essentialist* position. However, overall the operation of

usability does make something of a concession to a *contextual* approach as it considers Player Characteristics and Game Characteristics both as "inputs" into the *potential* for the players' enjoyment. This potential is then modified by the usability of the game to produce *actual* enjoyment. In this perspective, usability is an enabling quality of the game that at best should be never noticed,

> "...usability is a modifier that can hurt the experience when it is bad"
> "Usability is a problem when it is bad but once usability is good then it is basically invisible and not the actual source of enjoyment"
> "...game usability is about keeping the interface of the game from intruding in the player's ability to experience the game."

In particular, the system-centricity is explicitly expressed by the restricted inventory of game elements that it considers:

- HUD interface.
- Controls.
- Start / options menus.
- Level design.
- Visual appearance of game elements (enemies, avatars, items).

Schaffer provides definitive statements describing usability for games,

> "Game Usability makes a game's interface as transparent as possible, as quickly as possible."
> "The role of usability is to make the interface become an extension of the player and disappear"

2.2.4 Usability in the Product Lifecycle

Related to this discussion of usability and user experience is the question of the role it plays in the product lifecycle. The emphasis placed by user experience is on understanding the situated contextual needs and value of the product for real users. When usability is treated as a hygiene component alone it functions as a device for the identification and elimination of problems, rather than as a productive tool for ideation. It can be used both during summative and formative evaluation stages, though focussing solely on problems means that it has little to add to the generative processes of formative evaluation. In contrast, the summative stage of evaluation is not necessarily intended to guide the iterative design of a product during development, as is the case with formative evaluation. Clearly the lessons learnt during a final summative usability evaluation can help inform the development of future products, by identifying both positive and negative aspects of the system. The restricted, hygiene-only view of usability still has an important role to play. By attending to summative-only evaluation the scope of this thesis allows much greater attention to be paid to this specific area without having to address the potentially more ambiguous and complicated issues of formative, generative evaluation. Later in this thesis a novel evaluation methodology is developed and validated with summative evaluations. The potential for the method to be used in formative evaluation is discussed in Chapter 8 (Conclusions)

2.2.4.1 Conclusion

This review of the literature has shown that there is no single, coherent use of the terms usability, playability, and gameplay in the literature. This could suggest something of a critical aporia, similar to that seen with the terms usability (Cockton, 2012), user experience, and flow. Ijsselsteijn et al. (2007) even go so far as to assert that,

> "A standard for game experience assessment, like the well-know ISO usability standards (ISO 13407 and ISO 9241-11) is not likely to emerge any time soon."

For the purpose of this thesis, the following distinctions will be used,

Gameplay.
: How the player plays the game. This thesis does not address emergent or subversive models of play.

Playability.
: The intended task-oriented use of the game.

Usability.
: Functional use of the game as system for achieving game tasks.

The next sections discuss the meaning of the term "problem", and present a definition of usability problems for games used within the scope of this thesis.

2.2.5 Usability Problem

This section unpacks the concept of a usability problem in detail, providing important insights that will be pivotal for the remainder of this thesis. In particular, the separation of problem cause and outcome will be employed in later chapters to understand the weaknesses of traditional usability evaluation methods, and to develop a novel methodology for first-person shooter game evaluation that improves on current methods.

2.2.5.1 Introduction

Lavery et al. (1997) offer a definition of a usability problem,

> "...an aspect of the system and / or a demand on the user which makes it unpleasant, inefficient, onerous or impossible for the user to achieve their goals in typical usage situations."

They also distinguish different aspects of the term "problem", differentiating between the point at which the problem occurs (the breakdown), its cause, and the subsequent outcome of the incident,

> "We define a breakdown as occurring when the user does not take an inappropriate step in the interaction".

Example breakdown types include,

1. User forming an inappropriate goal.

2. User selecting an inappropriate action.
3. User not perceiving the feedback.
4. User misinterpreting the feedback.

Breakdowns are something that the user did, and causes are the design faults that result in the breakdown. Note that only item 2 is an observable action, the other s are cognitive processes that can only be verified by the player, otherwise they must only be inferred by the evaluator.

Capra (2006) defines a usability problem as being a problem experienced by the user, which is caused by an interaction flaw. This again emphasises the separation of cause and consequent outcome experienced. However, the definition of interaction flaws is not provided.

Nielsen (1994b) gives a definition of usability problems as,

> "...any aspect of a user interface that is expected to cause user problems with respect to some salient usability measure (e.g., learnability, performance, error rate, subjective satisfaction) and that can be attributed to a single design aspect ..."

The publication gives examples of problems found in a word processor that illustrate the definition of the term. For example, one problem was listed as dealing with users learning the standard cut/copy/paste commands. In this single problem, two situations were described,

1. The commands only worked when some text was already selected.
2. The copy command did not produce any feedback; some users were not sure whether the command had worked.

Nielsen's definition of a usability problem, then, implies that multiple different causes, breakdowns and outcomes related to a single design aspect should all be combined into a single problem. While ordering problems by design aspect seems like a reasonable approach, it may be more useful to separate each of the distinct issues with the design in order to better analyse, report and understand the problems. This idea is discussed in more detail elsewhere in this literature review, and is applied in Chapter 6 (The Playthrough Evaluation Framework) as part of the specification for a novel methodology, "playthrough evaluation", proposed by this thesis.

Welie et al. (1999) asserts that in general,

> "...usability problems are caused by a mismatch between the users' abilities and the required abilities that the system enforces on users."

This is similar to Norman's "Gulf of Execution", as indicated by Hartson (2003),

> "Mismatches between the designer's model and the user's view of this mapping contribute to the well-known Gulf of Execution."

However, rather than describing the actual cause of the problem, it describes the difference or space between problem and non-problem (the "gulf"). i.e., the mismatch between the user and system is a description of a consequence, which will result in an interaction breakdown,

but which in itself does not help the evaluator understand what it is about the system and / or user that caused the mismatch. For example, the designer may have taken for granted some implicit knowledge about how to use an interface, where to find a particular widget, or the users' ability to parse and understand the information presented to them.

Bolton and Bass (2010) addresses "erroneous human behavior" and cites the seminal work of Hollnagel (1993a,b). This is in the context of the latter's pioneering work on "erroneous actions", and in particular the distinction between their phenotypes (manifestations) and genotypes (causes). Both Hollnagel (1993b) and Lavery et al. (1997) observe that causes and outcomes are often referred to as "problems" or "errors". Hollnagel (1993b) defines "erroneous action" as,

> "...a certain type of action without implying anything about the cause ... an action which fails to produce the expected result and which may lead to unwanted consequences."

A. P. O. S. Vermeeren et al. (2002) paraphrase Lavery et al. (1997),

> "A usability problem is an aspect of the system and/or a demand on the user, which makes it unpleasant, inefficient, onerous or impossible for the user to achieve their goals in typical usage situations."

> "...a cause, a possible breakdown in the user's interaction, and an outcome, all of which happen in a context."

> "...within a specific context (e.g. user context, interaction context, task context), some cause (e.g. a design fault), may lead to a breakdown in the interaction (e.g. the user selecting an inappropriate action). This in turn may result in some undesired outcome (in terms of behaviour and/or performance; e.g. the user's task fails, the quality of the work suffers, the user becomes irritated)."

> "...the word 'breakdown ' will be used to include dialogue failures as well as knowledge mismatches."

The theoretical basis for usability and problems in this thesis is informed by these definitions and explored in more detail later in Chapter 6 (The Playthrough Evaluation Framework).

2.2.6 Usability Problem Discovery

This section provides an overview of the approaches taken to detecting usability problems. Several methods are considered, especially heuristic evaluation, user testing, and a number of more structured approaches such as cognitive walkthrough. The strengths and weaknesses of the methods are considered, particularly with respect to the question of reliability. Conclusions are presented with recommendations for a reliable approach to problem discovery for video games.

Problem discovery refers to the ways in which evaluators determine whether a problem exists or not. Given observation of a user test session, it is the ways in which evaluators notice and recognise that a problem has occurred, based on the interaction, or behaviour of user or system. Alternatively, in an expert inspection, it is the ways in which the expert

evaluator notices and recognises that a potential problem could occur, based on an analysis of the system and an understanding of the expected user behaviour.

Problem detection precedes problem classification, cause or outcome analysis, impact and frequency severity rating.

In addition to examining several different methods, Cockton, Woolrych, Hall, et al. (2003) propose four different approaches to problem discovery. Ordered by increasing planning and control effort for the evaluator they are:

- System scanning.
  An informal, freeform approach where evaluators do not follow a specific strategy. This freeform detection does not guide the evaluator at all, but rather just relies on their own informal, subjective and internal expertise to notice and recognise problems.
- System searching.
  Each element in the system is considered in turn, and analysed for problems.
- Goal playing.
  The evaluator plays a specific role in order to achieve a specific goal, but without using a formal procedure.
- Method following.
  User-goal oriented evaluation, but where the evaluator follows a formal procedure.

These terms are used through the remainder of this literature review, especially in Section 2.3 (Evaluation) and Section 2.6 (Heuristic Evaluation) where the literature is examined and described with these terms.

2.2.6.1 Problem Detection Criteria

Problems are often first noticed by observing their consequences in terms of usability outcomes. In the model described by Lavery et al. (1997) outcomes come in three types,

1. Behavioural:
   - User forms an incorrect goal.
   - User does not select the correct action.
   - User incorrectly interprets the system's response.
   - User stops, and fails the task.
   - User tries incorrect actions.
2. Performance:
   - Time on task increased.
   - Task failure.
   - User recovered from breakdown.
3. Preference:

- This Satisfaction outcome was only proposed, but not detailed in the published paper. Presumably it would deal with cases where there were no usability outcomes impacting Effectiveness or Efficiency, but rather where the user simply did not like the interaction.

Note that behavioural outcomes include both objectively observable interactions, such as the user trying an incorrect action, and also purely cognitive phenomena that only the user can verify, such as forming an incorrect goal. The evaluator may be able to infer these outcomes on the basis of other observable behaviour, however.

In this model, causes are either design faults, or unfulfilled knowledge requirements of the user. A cited example is a button that is not salient for the user. A consequence being that the user is unable to identify the correct action to succeed in their goals.

2.2.7 Event Indicators

Jacobsen et al. (1998b) describe a user test where three criteria were used to indicate a problem:

- User appears stuck for 3 minutes or longer.
- User gives up task.
- User stops thinking out loud.

Later in the same publication, the following 9 criteria are used (Jacobsen et al., 1998b) to detect usability problems:

1. The user articulates a goal and cannot succeed in attaining it within three minutes.
2. The user explicitly gives up.
3. The user articulates a goal and has to try 3 or more actions to find a solution.
4. The user produces a result different from the task given.
5. The user expresses surprise.
6. The user expresses some negative affect or says something is a problem.
7. The user makes a design suggestion.
8. The system crashes.
9. The evaluator generalises previously detected problems into a new problem type.

Some of these user responses may need to be treated differently in the context of a game play evaluation. For example, given that players are expected to struggle and fail to a certain degree, surprise, negative affect, and design suggestions are not necessarily indicators of gameplay problems. However, there is still scope to apply these general principles to the basic usability aspects of first-person shooter games. For example, if a player tries to use a menu system, we can assume that the designer had intended it to be easy to use. In this case, the criteria above may still be appropriate. Exceptions are obvious though, for example if a player is surprised by a sudden attack from a hidden non-player character, this would

generally not indicate a problem in a game evaluation. Even a naïve interpretation of negative affect, when directed at the difficulty of a game scenario, should be carefully considered by the evaluator and not necessarily recorded as a problem *per se*. In some cases the player will only experience the pleasure of successful mastery after first experiencing the dissatisfaction of failure. Frustration and anger can, at times, be an important and even necessary component for an overall positive experience.

2.2.8 Theoretical Models of Interaction

Barendregt et al. (2006) considers the taxonomies of Rasmussen (1982) and Zapf et al. (1992), separating functionality, usability, interaction and inefficiency. Citing Pagulayan et al. (2003), functionality problems are not considered to affect video games as these types of problems are defined as the "mismatch between tasks and the program" where tasks are implicitly extrinsic, and video games are considered to "only have internal goals and no external goals or tasks."

From the taxonomy of problems, this only leaves usability and inefficiencies for consideration in game evaluation. Usability problems are defined as occurring "when the functionality of a program is sufficient for its execution but there are still problems" due to "a mismatch between the user and the computer program." Inefficiencies are defined as "when the user is successful in reaching a goal that could have been reached more easily because the system does not make this more efficient way clear to the user."

The taxonomy in Zapf et al. (1992) proposes "knowledge problems" for cases where the user is unable to carry out their task due to unclear explanation from the program. In terms of action regulation, the taxonomy proposes that "Thought problems" occur when users develop inadequate goals or plans due to misunderstanding the program. "Memory problems" are simply when the user has forgotten aspects of the program. "Judgement problems" deal with the user's misunderstanding of feedback from the program. In terms of action patterns, the taxonomy proposes that "Habit problems" occur when users perform the correct action in the wrong situation. "Omission problems" are when users do not compete a well-known subplan. "Recognition problems" deal with users not noticing feedback from the program. Finally the taxonomy proposes "Sensoriomotor problems" for motor-skill or coordination issues. As far as inefficiencies are concerned, the taxonomy identifies two types: inefficiency due to a lack of knowledge, or due to incorrect habit.

Norman's theory of action (Norman, 1986) is an influential model for how users interact with computer systems, developed from Hutchins et al. (1985) which examined direct manipulation devices and identified two important qualities that contribute to a feeling of directness,

1. The relationship between the user's intentions and the facilities of the system.
2. The relationship between the experience of objects in the system and the real world experience of those objects.

These qualities provided the foundation for the more general "Gulfs" that can cause problems in general interaction,

1. The Gulf of Execution between a user's goals and their knowledge of how to achieve them with the system.

2. The Gulf of Evaluation between a user's goals and the description of the system's state.

2.2.8.1 User Action Framework

The user action framework is a hierarchical tree structure for the classification of interaction problems. It is based on Norman's stages of interaction (Norman, 1986), and has been widely cited by the research community as tool to analyse usability problems (Capra, 2006; Chattratichart and Lindgaard, 2008; Howarth, 2007; Thompson, 1999).

It is oriented towards traditional Windows, Icons, Mouse, Pointer domains, with nodes referring to the manipulation of "UI objects" for example. Most first-person shooter video games do not display a conventional user interface in these terms, but rather feature a head-up display to present certain status information (such as character health, ammunition, etc.) over a dynamic 3D environment containing numerous objects. Many of these objects may be interactive, but without an explicit function or task-related significance.

One of the challenges involved in using a classification scheme such as the user action framework is to adapt the terminology to be meaningful for the new domain. A typical use case for the user action framework might be to consider the "widgets" that a user could use in a desktop accounting application, such as buttons, menus, sliders, tick boxes, data entry fields, etc. Additionally the user is likely to only interact with the application using a keyboard and mouse. Each mouse button or key has a consistent response across most (if not all) applications within this domain. In contrast, interaction with "game objects" is usually enacted through a dedicated physical controller, which uses analogue and digital buttons, analogue joysticks, and analogue or digital triggers. Buttons may have contextual meanings, which can involve the current state of the game (including the location of the player character, their available inventory, ammunition status), or have particular meanings based on the sequence in which they're used.

In a traditional Windows, Icons, Mouse, Pointer environment it makes sense to talk about an interaction in terms of user interface objects, such as clicking on the File menu, selecting the Save As entry, typing in a file name, then clicking the OK button. In contrast, a typical "User interface" interaction in an first-person shooter game might involve a pushing the left analogue stick to the right, while pushing the right analogue stick to the left. While in combat, this could result in the player character "circle-straffing" around an enemy in a counter-clockwise direction, while keeping the enemy in the centre of the screen (the left analogue stick moves the player character to the right, while the right analogue stick counter adjusts their point of view, such that the relative direction "right" is continually rotating counter-clockwise, describing the tangent to a circle around the enemy). While circle-straffing, the player might cycle through their currently available weapons by pressing the "Y" controller button. This is a form of modal interaction, where the consequence of pressing "Y" depends on the player character state with respect to the weapons currently held in their inventory. e.g., if the character is currently equipped with a pistol, and has a shotgun available, then pressing "Y" will switch between these two. However, if they additionally held a rifle, then the weapons would cycle through in sequence, e.g., pistol - shotgun - rifle - pistol - etc. Finally the player would pull the primary trigger to fire their currently equipped weapon.

In this example, the "user interface objects" involved arguably include the physical controller components (sticks, buttons, triggers) as would be expected, but potentially also the game environment (around which it may not be possible to circle strafe, for example if the player character and enemy are in a small corridor), certainly the enemy, the weapons used and their position in the inventory cycle. The distinction between game environment and game entity (such as player character or non-player character) diminishes further if we consider cases where terrain is hostile (causing damage by proximity to acid, spikes, or falling long distances) deformable or interactive in pursuit of a game goal (piling up rocks to reach a higher level), or can be used for tactical advantage (hiding behind cover).

The user action framework primarily deals with cognitive and sensory issues. While there are a large number of these kinds of challenges and problems in first-person shooter games, they are also unique in that they require physical skills to be developed too. It is expected that some amount of challenge will be involved in gameplay, but the user action framework is designed from the point of view of minimising challenge and maximising the traditional usability issues of Effectiveness and Efficiency. As such, the user action framework needs to be adapted for use with games. It does however show a great deal of potential, and has been iteratively developed by many researchers across many projects. As such, aspects of it are used in Chapter 6 (The Playthrough Evaluation Framework) for the development of a novel usability evaluation method for evaluating first-person shooter games.

2.2.9 Usability Problems in Games

Similar to Ijsselsteijn et al. (2007) observing that standards for game experience assessment are unlikely to be developed, Cockton and Lavery (1999) conclude their paper by asserting that,

> "...there is no universal definition of a 'usability problem', nor can there be one."

Nonetheless, within the scope of this thesis, a usability problem is defined as,

> An undesirable consequence of the interaction between player and game, caused by a mismatch in the relationship between design decisions and player's ability, producing a cognitive or physical breakdown where the interaction does not proceed according to the designers' or player's expectations, where the player is unable to use the game to achieve the goals expected by the game, resulting in an undesirable outcome experienced by the player in terms of the usability aspects of efficiency, effectiveness, or satisfaction.

These aspects can be undesirable in overly positive as well as negative ways. e.g., if the player is able to overcome the game's challenges too easily then these can be experienced as efficiency or effectiveness being undesirably high.

2.3 Evaluation

The questions posed by this section include the following:

- What are the criteria for comparing between usability evaluation methods?
- What experiment designs are used to compare evaluation methods?
- Which methods have been compared, and what are the general findings?

This section concludes by reiterating these questions and summarising the approaches found in the literature.

2.3.1 Evaluation Relative to Product Lifecycle

Usability evaluations are usually divided by where they occur relative to product development and release. This also has implications for how to conduct the evaluation, including which method to use, and who should be the evaluator.

Scriven (1967) defines two types of evaluation relative to the product lifecycle:

- Formative.
  Conducted during the development of a system, particularly to feed back to the developer.
- Summative.
  Conducted on a completed product ready for the user.

And two focusses for conducting evaluations:

- Intrinsic.
  Evaluation of the system and design in itself.
- Payoff.
  Evaluation of the effect of using the system.

Combinations of these lead to four possible cases:

- Formative intrinsic evaluation.
  For example, expert or heuristic evaluation studies that use prototypes.
- Formative payoff evaluation.
  This case is used to judge the interim usability of a system.
  e.g., user testing of prototype systems.
- Summative intrinsic evaluation. Final judgement of the system's design. For example, an expert evaluation of a finished product.
- Summative payoff evaluation.
  Final judgement of the usability of the system.
  e.g., a user test of a finished product.

He also recommended that formative evaluations be conducted by someone close to, or embedded within, the development team. Summative evaluations, which emphasise unbiased, and objective analyses, should instead be conducted by an independent and external party.

Hartson et al. (2001) adopts Scriven (1967)'s terminology to describe the difference between formative and summative evaluations. Those that occur while the product is still in development, with the intention of informing redesign prior to release, are called *formative*. Those whose purpose is to provide a summary of a product, usually after it has been finished, are called *summative* evaluations. Formative evaluations tend to be more qualitative, and summative more quantitative. Summative evaluation is regarded as requiring more formal and rigorous testing, including quantitative analyses. M. Rosson and J. Carroll (2002) also cite Scriven (1967), distinguishing formative evaluation as taking place during the design process.

Gram and Cockton (1996) define summative evaluation as,

> "...structured and planned evaluation of the finished product by usability specialists, with measurement against required targets." (p.67)

This is in contrast to the implied normal, formative, early, iterative evaluation (p. 66). Similarly, the UXPA/UPA (User Experience Professionals' Association - formerly the Usability Professionals' Association) defines formative evaluation as,

> "...testing with representative users and representative tasks on a representative product where the testing is designed to guide the improvement of future iterations."
>
> Theofanos and Quesenbery (2005)

where the purpose is,

> "...to improve a product during its design and development."

Note that this definition excludes any method that does not include representative users, tasks, and products, for instance, heuristic evaluation, cognitive walkthrough, surveys, focus groups, etc. In contrast to formative, summative evaluation is defined as occurring at the end of development. Hix and Hartson (1993) defines formative evaluation as

> "...evaluation of the interaction design *as it is being developed*, early and continually throughout the interface development process." (p. 284)
>
> "...performed several times throughout the process." (p.285)

This contrasts to summative evaluation as,

> "...evaluation of the interaction design *after it is complete*, or nearly so." (p. 284)
>
> "...usually performed only once, near the end of the user interface development process." (p. 285)

Furthermore they suggest a different quantitative approach for summative evaluation,

> "...formative evaluation, the mainstay of usability evaluation, is not to be confused with what is often thought of as typical human components testing – for example, controlled hypothesis testing of an m by n componential design with y independent variables, complete with quantitative data, statistical analyses, and numeric results."

Wixon (2003) argues that the success of formative evaluation methodologies critically includes social, political, practical and business components. Furthermore, formative usability engineering should be conducted from the perspective of the engineering method rather than the scientific method. As such, a case study model is advocated rather than an experimental approach to comparing usability evaluation methods which has been used in much of the literature. The underlying assumption of an experimental approach is that the most efficient usability evaluation method can be determined in terms of the number of problems found as a function of the number of participants used. In contrast, the goal of the engineering perspective advocated is,

> "...to produce, in the quickest time, a successful product that meets specifications with the fewest resources, while minimizing risk."

Sensitivity to project logistics is of paramount importance, and ultimately, the purpose of the method is for problems to be fixed, not just found. Lab-based approaches to formative evaluation are flawed when they attempt to isolate the method from the process in which the method is intended to be used. As such, comparisons between methods used in these conditions lack a real world context to make their results meaningful. Similarly, Cockton (2012) discusses how usability methods in themselves only constitute part of the spectrum of usability work. A sensitivity to context, and the limitations of lab based exercises is also drawn out in Law et al. (2009).

2.3.1.1 The Relationship Between Type of Method and Product Lifecycle

Gray and Salzman (1998) bring into the discussion a distinction between types of usability evaluation method. Empirical approaches, such as methods employing various forms of user testing, are able to measure what they refer to as "payoff" usability. This is usually expressed as performance metrics such as time-on-task, number of errors, etc. In contrast, analytical approaches, such as heuristic evaluation, inspect the "intrinsic" properties of a system and attempt to make inferences about potential usability problems that could occur.

Cockton (2012) similarly differentiates usability evaluation methods as being analytical (based on inspection or examination of the system and its potential) or empirical (based on actual use). Analytical methods in turn are distinguished as being either more system-centred, such as heuristic evaluation, or interaction-centred, such as cognitive walkthrough. System-centred approaches are said to attend only to the properties of the system, whereas interaction-centred approaches also consider the context and users. While analytical methods emphasise the *causes* of good or bad usability, empirical methods tend to emphasise the *effects*.

Types of Method

Nielsen (1994c) describes four general approaches to evaluation,

- Empirical (e.g., user test).
- Automatic.
- Formal.

- Informal (e.g., heuristic evaluation).

Informal expert evaluations rely on the skill of the specific evaluators involved, and sometimes address purely superficial aspects of the interface. In contrast, more formal, systematic methods attempt to fully specify a procedure to produce reliable and comparable results regardless of the evaluator (Matera et al., 2002). These approaches can be seen as embedding a significant portion of the success or applicability of the methodology either in the evaluator, or in the procedure. In the former case, deferring a greater degree away from the methodology *per se* and embodying it in the evaluator means that it becomes difficult to examine and compare the methodology *in vacuo*. Rather than an evaluator relying on an external object to guide them (i.e., the written procedure), they instead have to rely on their built up implicit experience, which may be largely inaccessible on a conscious level and potentially difficult to express.

The Relationship Between Analytical and Empirical Methods

> "...experts identify an interface error and *predict* that it will cause user problems, whereas for end-users we identify the symptom and then must *infer* the cause or more general problem. The experts are looking for causes of error and predicting effects. The end users encounter effects, from which we infer causes."
>
> Doubleday et al. (1997)

Empirical approaches are particularly well suited to summative testing of working products with actual users and representative scenarios, as the real payoff usability is readily apparent. Payoff usability metrics based on early prototypes are unlikely to be representative of those seen in a final product, but may still have some use for identifying indicative potential problems.

The main benefit of analytical approaches is that they do not require user involvement, and so can be applied during early prototyping during formative assessment. Clearly they can still be applied to a final, working product, but in this situation real users could be observed performing representative tasks with the the actual system. To not make use of this resource would be at best a waste, and potentially could result in invalid conclusions if the analytical method is not validated against empirical data.

Empirical methods are better suited to identify problem consequences or outcomes, but tend to lack the resources to specify the cause of a problem. Analytical methods have the opposite stance, specifying causal design features but not being able to make necessarily strong predictions about consequential outcomes. This relationship between intrinsic features and payoff was noted to be weak in all of the studies examined by Gray and Salzman (1998). They argue that a systematic approach is required that would relate intrinsic features and payoff consequences.

L.-C. Law and Hvannberg (2002) compared heuristic evaluation and user testing, and cite Gray and Salzman (1998) to conclude about heuristic evaluation that,

> "...this analytical usability evaluation method is not apt for making 'forward inference from intrinsic feature to payoff"
>
> "...the predictive power of heuristic evaluation is moderate."

2.3.1.2 Playthrough Evaluation Reconciles Empirical and Analytical Resources

The novel method presented later in this thesis, playthrough evaluation, addresses evaluator resources through both empirical and analytical phases of evaluation. The identification and reporting of empirical problems is expressed using a terminology derived from a detailed taxonomy of intrinsic design features. The same unified terminology is used to guide a systematic evaluation of intrinsic design, as well as to transcribe incidents observed during empirical testing.

2.3.2 Game Evaluation Lifecycle

In a formative evaluation, the purpose is to inform the ongoing design of a system. In this case, contextual issues such as the practicality of the recommendations are important, as pointed out in the preceding literature. What's more, in order to provide recommendations and design solutions that are of practical use, the evaluator needs to be not only an expert in human-computer interaction but also an expert in designing the domain being analysed.

Video game design is a formative, creative activity, whereas game evaluation is more of an analytical, summative activity. This may be more so the case than with traditional products which are concerned more closely with functional requirements than aesthetics.

As this thesis deals with a relatively new domain, it seems prudent to focus primarily on summative evaluations. Attempting to address formative evaluation on a system that is still under development would introduce too many additional uncontrollable and unknown variables. For example, there has been scant research on how representative early evaluations can be for such complex, multi-modal and interrelated systems.

During a discussion at a user experience workshop in 2010, one game designer from the Climax Portsmouth game studio described how an early prototype level for their game appeared to work well in preliminary user testing. However, when the level was completed with final quality textures, players had difficulty visually parsing the scene due to the differences in textures used. This resulted in problems with navigation, and hence the overall player experience was negatively impacted. Formative evaluation using the unfinished assets would not have been able to predict the impact this would have on the usability of the final game.

Once a reliable framework for summative usability evaluation has been established, it will then be possible to extend this work to consider the more ambiguous area of formative evaluation. The structured and reliable testing framework described in this thesis (particularly in Chapter 7 (Testing Playthrough Evaluation)) is designed for summative evaluation, so does not directly address the specific requirements of testing early stage prototypes. Given the dynamic nature of development, more lightweight methods such as heuristic evaluation with less emphasis on reliability may be more suitable during formative stages of the game evaluation lifecycle.

Most of the literature on video game usability evaluation employs formative techniques, such as heuristic evaluation, but in a summative manner once the product is finished. The details of these cases are explored in more detail in the sections on Section 2.6 (Heuristic Evaluation). At this stage it is important to appreciate why this approach is problematic. Simply put, formative usability evaluation methods do not provide the rigour of summative evaluation,

and their value should be interpreted in terms of the ability to actually improve real products.

2.3.3 Evaluating Videogame Usability

Usability is a foundation and potential barrier to gameplay. Poor usability may negatively impact on what would otherwise be good gameplay, but good usability does not necessarily contribute to good gameplay, however. On the other hand, poor usability can be worked around by player competencies. For example, poor control layouts can be mastered although they may require more attention or practice from the player. Good gameplay can serve as the motivation to overcome poor usability. For example, the *Grand Theft Auto* series has often been criticised for unintuitive, complex, and awkward control schemes. Evaluating the product purely in usability terms, this would be considered a serious failure. Nevertheless it is critically and commercially acclaimed for its engaging gameplay, despite these more basic shortcomings. Players are willing to overlook these issues, and find coping strategies to work around the initial problems.

The definitions presented here are particular to this thesis, as there are few broadly accepted definitions and distinctions between these terms in the literature.

Human-computer interaction provides formative usability methods for the design, development, and evaluation of the user experience, with a particular attention to issues of use. Ludology provides formative playability methods for the design, development, and evaluation of the player experience, with a particular attention to issues of play.

Formative evaluation is concerned with the design as *potential*, whereas summative evaluation is concerned with the *final implementation*. Summative usability evaluation is concerned with the actual implementation of a system from the perspective of use.

Summative usability evaluation is the most concrete form of evaluation for video games. It looks at actualised implementations rather than potential designs, and with basic questions of use rather than the more nebulous properties of play. As such it is the most appropriate starting point for future work.

The difference between formative and summative evaluations are further exacerbated when considering the differences between usability and playability. Formative playability methods are available for rapidly and iteratively developing and evaluating game mechanics, dynamics, and their effects on the player experience. For example, Järvinen (2008) provides a comprehensive set of tools for this purpose. However, it does not specialise on the means to assess the usability of a game. In contrast, heuristic evaluation does specialise in rapid evaluation of a system's usability, regardless of whether that system is a game or a productivity application, for example.

The main purpose of formative evaluation is to in*form* design, especially in an iterative way. Summative evaluation produces a *summary* of the existing design. While a summative evaluation needs to understand and report problems, a formative evaluation is in some ways more demanding. In order to usefully contribute to the ongoing development of a system, it is not sufficient to merely point out existing problems, but rather to provide solutions too.

This thesis addresses the fundamental question of how to evaluate a game. Only once evaluation has been appropriately performed should redesign be applied. Errors introduced during the evaluation stage could negatively affect new designs, fail to resolve existing problems,

and introduce new issues.

2.3.4 Informing Design Requires Domain-Specific Design Expertise

Norman discusses some of the qualities that make a good system design, giving several examples including the UNIX operating system, spreadsheets, and the *Pinball Construction Set* (PCS) (Budge, 1983). He cites the PCS as an example of good design, particularly in terms familiar to usability. However, he also makes an important observation regarding design. In this case, using the PCS as a tool to design new pinball games,

> "Much as I enjoy manipulating the parts of the pinball sets, much as my 4-year-old son could learn to work it with almost no training or bother, neither of us are any good at constructing pinball sets."
> Norman (1986) p. 51.

Despite being an expert in human-computer interaction, and commenting that the game facilitates a very usable experience, Norman recognises that he lacks the ability to design a good gaming experience. Cockton (2012) cautions against giving too much authority to methods alone, pointing out that methods are just one aspect of usability work, and that evaluators' expertise is a necessary component. This is especially true in formative evaluation, where, as recommended by Scriven (1967), evaluators are embedded within the development organisation so that they fully understand the development constraints of the project. In the case of evaluators who do not have game design experience, however, proposed changes based on faulty evaluation may in fact result in worse game design decisions being made. While there are some human-computer interaction experts who also have professional game design expertise, the number is currently very small, and most game development organisations do not yet employ usability evaluators even on a temporary basis (McAllister and White, 2010). With the advent of new supporting businesses such as Vertical Slice, which provided outsourced evaluations specifically for the video game industry, awareness of the benefits may improve and spread throughout the industry. At this stage, however, in the rare occasions when summative evaluation is conducted, it is likely to be performed by evaluators who are not also game design experts. This is true even of the new specialist companies like Player Research[2], which have no experience in game design or other areas of game development.

2.3.5 Summary

Most applications of user testing employ a freeform, system-scanning approach to problem detection, where evaluators simply use their observational skill to notice, recognise, classify problems, analyse or infer their underlying cause, and then rate them for severity.

Some applications of heuristic evaluation take a heuristic-driven approach, where evaluators consider each heuristic in turn and attempt to match qualities of the interface or user interaction against the heuristic description. This has the potential drawback that evaluators will not notice issues that are not described by any of the heuristics, and so this may result in a decrease in the metrics for coverage or thoroughness. However, the additional structure it

[2] http://www.playerresearch.com/

presents may increase reliability as evaluators are more likely to all employ the same technique to locate and classify issues.

The most formal methods such as cognitive walkthrough assume a priori that an optimal procedure exists for tasks. Furthermore these procedures are expressed at the very lowest level of interaction, comparable to the keystroke-level model (Card et al., 1980) where individual operations such as key strokes and mouse clicks are prescribed. While this level of detail is reasonable for simple traditional systems, it would be excessive for interaction-dense first-person shooter games.

Video games offer a much more emergent, dynamic and complex environment than traditional Windows, Icons, Mouse, Pointer domains. The level of detail used in formal approaches such as cognitive walkthrough would be infeasible for such an interaction-dense environment. The emergent nature of most games is such that it is not possible to define a correct or even optimal sequence of interaction events with the granularity of the keystroke-level model. However, it may be possible to construct patterns of interaction at higher levels as first-person shooter games do tend to be goal oriented.

Several studies avoided the challenges of problem detection by presenting evaluators with a list of pre-defined problems, with inter-evaluator reliability computed based on this fixed list. This approach may be useful for falsification testing, but cannot give an accurate measure of the method's reliability in a general sense. In a real setting it will usually not be possible to define an exact set of known usability issues in advance. Indeed, if this were the case then evaluation would be purely for the sake of validating a method rather than for the more natural purpose of summarising the usability of a system.

A method for detecting problems in first-person shooter games would need to fulfil the following criteria:

- Be flexible enough to respond to the dynamic nature of each different playthrough.
- Be reliable enough that evaluators can agree on what the problems are.

The terminology and definitions from Scriven (1967) inform the remainder of this thesis. In particular the novel method developed later in this thesis, playthrough evaluation, is designed to address the following points:

- Summative evaluation.
- Independent evaluators, external to the development team.
- Attention to both intrinsic and payoff qualities.
- Theoretical grounded.
- Measured by reliability.

These findings are used to inform the studies and experiments used later in this thesis, and are referred back to in their individual chapters.

2.4 Evaluator Effect

The evaluator effect is a symptom observed when evaluators come to different conclusions regarding the same evaluation. For example, L.-C. Law and Hvannberg (2002) conducted a comparison of user testing and heuristic evaluation. In their analysis they noted substantial differences in usability problems found,

> "The idiosyncrasy of individual test participants, for instance, their technological knowledge and even personal aesthetic preference, affects whether a UP is named."

This influences their later work (E. L.-C. Law and Hvannberg, 2004a) in which they define a novel term, the "user effect". This has a more general meaning dealing with the capacities of users in general to capture usability problems of a system. This can be applied to both users as evaluators as well as end users of a system. Likewise, Gray and Salzman (1998) define the wildcard effect,

> "...people who are significantly better or worse than average and whose performance in the conditions of the study do not reflect the UEM but reflect their Wildcard status."

This section describes the literature on these subjects, showing how widespread the problem is even with relatively simple usability evaluation in traditional domains. In more complex environments such as first-person shooter games, the issue is compounded due to the complexity of the task involved, the freeform, dynamic, and emergent nature of the experience as opposed to the more straightforward and often linear task structure, and the rather more nebulous qualities of play rather than use.

Hertzum and Jacobsen (2001) conduct a meta-review of 11 usability evaluation studies, comparing cognitive walkthrough, heuristic evaluation, and think aloud. The purpose of their study was to explore the evaluator effect which they define as

> "...differences in evaluators' problem detection and severity ratings."

They use it to discuss measures of reliability,

> "...the extent to which independent evaluations produce the same result".

They report that the evaluator effect occurs for all three methods, and in diverse conditions regardless of whether evaluators are novices or experts, for problem detection, severity assignment with issues that are serious or cosmetic, and for both simple and complex systems. Three primary sources for this effect are identified as:

- Goal analysis.
- Evaluation procedures.
- Problem criteria.

They comment that,

> "...differences in the evaluators' thresholds regarding when a difficulty or inconvenience becomes a problem are generally not regulated by the UEMs and must be suspected to contribute considerably to the evaluator effect."
>
> Hertzum and Jacobsen (2001)

Their study found that average agreement rates varied widely, from 5% to 65%, and none of the three usability evaluation methods considered produced significantly more reliable results than the others. They conclude that this effect is a consequence of the inherent subjective judgement required by the evaluators, and recommend three key points,

- Be explicit on goal analysis and task selection
- Involve an extra evaluator, at least in critical evaluations
- Reflect on your evaluation procedures and problem criteria

In regards to the first point, they call for precise operational definitions of usability problems. Playthrough evaluation, the novel methodology described in Chapter 6 (The Playthrough Evaluation Framework), provides this by defining interaction patterns that describe events involved in an interaction. Influenced by the recommendations of Cockton et al., these events are categorised by context, breakdown, and outcome.

The final point about reflecting on procedures and criteria is also well addressed by playthrough evaluation. One of the main benefits of this method is the explicit definition of procedures that can be analysed, critiqued, and improved by the research community.

2.4.1 Stages of Evaluation

This section addresses the stages of evaluation used in any usability evaluation method, and their potential impact on the evaluator effect. Evaluation consists of the following stages:

- Problem discovery.
- Problem analysis.

2.4.1.1 Problem Discovery

Problem discovery describes how the evaluator initially comes to regard a particular incident or area of the system as being of interest. This is prior to any kind of classification of the kind of problem, causes, severity, merging of duplicates, filtering, or other analysis. This process can roughly be characterised as formal or informal. Informal approaches tend to be the norm, and leave it up to the evaluator to decide when an incident should be considered for analysis and reporting, while more formal approaches such as cognitive walkthrough define strict criteria for "correct" behaviour, which makes it straightforward to notice erroneous deviations.

Validity of Problems Discovered

In addition to their concern for validity, Gray and Salzman (1998) also contributes terminology to describe the types of usability problem discovered during an evaluation, particularly when dealing with methods that *predict* that an issue will be a problem for real users.

- Hit:
  The evaluation claims that a specific issue is a problem, and it really is a problem for real users.
- False alarm:
  The evaluation claims that a specific issues is a problem, but it is not really a problem for real users.
- Correct rejection:
  The evaluation claims that a specific issue is not a problem, and it really is not a problem for real users.
- Miss:
  The evaluation claims that a specific issue is not a problem, but it really is a problem for real users.

These terms are later added to in Cockton, Woolrych, Hall, et al. (2003); Woolrych et al. (2004) with:

- Genuine miss.
  A real problem was not identified by the evaluation

In general the truth or false value of a prediction is determined by whether the issue is reported in user testing.

Resources for Problem Discovery and Analysis

Cockton et al. (2004) conducted heuristic evaluations and identified four different "Discovery Resource" approaches that evaluators employed during the problem discovery stage of their evaluations:

- System Scanning.
  This is the most common approach used in unstructured and informal methods like heuristic evaluation and user testing, where evaluators simply note whatever catches their eye. No particular strategy is used.
- System Searching.
  A system-structured approach to analysing the product. Evaluators systematically review the design elements of the product.
- Goal Playing.
  Conducted by enacting an unstructured, user-centred scenario. Requires a little domain knowledge, but produces well-grounded, valid predictions as the evaluators perform representative tasks.
- Method Following.
  As with Goal Playing, but structured to use a formal method.

In their studies, the System Scanning approach resulted in lower rates of problem elimination, and higher false positives. They concluded that this quick and easy scanning "method" found problems easily, but which were easy to misinterpret and report erroneously. They

note that the most effective strategies employed more than one discovery resource, as different perspectives often allows evaluators to reject false positives. These issues would have otherwise been mistakenly allowed through to the reporting stage.

The novel playthrough evaluation method developed later in this thesis (Chapter 6 (The Playthrough Evaluation Framework)) defines ways to use each of these resources, emphasising Goal Playing and System Searching.

Implicit and Explicit Problem Detection

The two most common ways of detecting usability problems are inspection and user test, though each is still subject to problems of reliability. Regardless of whether inspection or user testing is used, problem detection can proceed in an explicit or implicit manner. With *explicit* inspection, the evaluator uses an inventory of the system features, tasks, or components, and considers each in turn. Cognitive walkthrough uses a very explicit form of problem detection, as it defines the tasks to be evaluated, and the specific operations needed to complete them optimally. Measurable usability criteria may be defined to determine the success or failure of each. A structured approach such as system-scanning would exemplify the explicit style of evaluation. In an *implicit* inspection, the evaluator conducts the evaluation in a more freeform, open manner until a particularly noteworthy feature, task, component or event is noticed. Heuristic evaluation is amenable to either explicit or implicit approaches to problem discovery, though most of the studies in the literature employ implicit expert inspection. In this form the evaluators interact with the system and use heuristics to help them consider when aspects of it could be problematic.

2.4.1.2 Problem Analysis

Independent evaluators each produce their own list of usability issues that they discovered, and often more than one evaluator will discover the same issue as another. In these cases the individual problem tokens that represent the same underlying problem are matched together. A final master list shows all of the different problems discovered by all of the evaluators, with duplicate matches discarded.

All studies perform some form of matching, albeit informal and not open to reflective analysis. Hornbæk and Frøkjær (2008) present a study exploring the issue in detail, and argue that this phase of evaluation is a key determinant of the evaluator effect. They defined the process of matching as,

> "...comparing usability problems found by different evaluators to assess whether they concern the same or different problems."

Their study found an average Any-Two value of 7.77 for the matching stage, which represents rather weak inter-evaluator reliability. To explain these findings they propose a novel term, the "matcher effect", as the difference between analysts in the matching phase of an evaluation.

Cockton and Lavery (1999) also discuss the problem of matching problems to one another, and cite three similar cases from an evaluation where it was unclear whether they should be

considered the same issue:

- The "Clashes" label does not afford selection.
- User selected "Tools" menu (not "Clashes" button).
- Users must know that the "Clashes" label is selectable.

They propose that better procedures will be necessary in order to ascertain whether these three issues all describe the same problem. This is particularly important with respect to evaluation predictions being compared to actual user test data. These three different forms of issue report each emphasise a different aspect of potentially the same problem, but seem to have been created through different kinds of evaluation. The first has a form typical of a cognitive expert evaluation which suggests a potential but not necessarily actual problem; the middle is an actual user test outcome observation based on the kind of optimal task path defined by cognitive walkthrough and similar methodologies; the third report does not mention a specific user problem but is rather more like a sensible design principle. The different forms that evaluation takes and the implications for them are discussed in further detail in Chapter 5 (Exploring Evaluation Resource Specificity). The overall goal of producing better evaluation procedures is generally addressed by the novel contribution of this thesis, playthrough evaluation, introduced later in Chapter 6 (The Playthrough Evaluation Framework).

Gray and Salzman (1998) examine the validity of usability evaluation, and note that reliability as a general concept has an impact throughout the evaluation process. In particular they identify problems of "instrumentation" as a potential threat to the validity of usability evaluation methods. This is defined as when evaluators "identify, classify, or rate the usability problem". In regards to the matching problem they recommend that,

> "Developing a common categorization scheme, preferably one grounded in theory, would allow us to compare types of usability problems across different types of software and interfaces."

Evaluator Effect in Heuristic Evaluation

Sim (2009) discusses the high degree of variability in evaluator performance of heuristic evaluation. He argues that it is a fundamental weakness of the methodology that there is no way to measure an evaluator's performance prior to conducting an evaluation. However, being able to repeat a formalised procedure would allow evaluators to be compared to one another.

Most heuristic evaluations involve informal stages where evaluators hold private discussions together to resolve differences. This aporia in the methodology means that even with the same product, it would not be possible for independent evaluators to follow the same procedure as one another, as a significant proportion of the methodology is deferred to the evaluators' private, informal discussions. Rather than being encapsulated within a formal procedure, any knowledge gained or discovered by a single group of evaluators is by-and-large lost for the rest of the research community.

Informality and the Methodology Effect in Problem Matching

Jacobsen et al. (1998b) describe a study where 4 evaluators observed and documented user test footage. Once all 4 evaluators had produced their reports, 2 analysts separately merged all of the problems into a summary list. No explanation was given for how the problems were merged by the individual analysts, in particular how issues were matched together, and how they were merged if the criteria violated were listed as different between evaluators.

Overall, they observed a substantial evaluator effect, as only 41% of severe problems were detected by more than one evaluator. This is related to their definition of the term "severe", which is any issue violating the following three problem detection criteria:

1. The user articulates a goal and cannot succeed in attaining it within three minutes.
2. The user explicitly gives up.
3. The system crashes.

Of the unique problems in the final list which were rated by the merging analyst as violating one of these three points, 73% were rated by at least one other evaluator as violating a different, non-severe criterion instead. They conclude by stating that it is not reliable to rely upon any one individual's evaluation of which single criterion has been violated.

Validity

Gray and Salzman (1998) discuss questions of validity in usability evaluation. They emphasise the threat to validity from "instrumentation", which involves stages of the process where evaluators "identify, classify, or rate the usability problems".

These stages of the process are often overlooked or conflated in most evaluation studies. Explicitly separating and defining procedures for each stage will facilitate more accurate insights into the value of a usability evaluation method. They suggest that,

> "Instrumentation problems can be avoided by treating the identification, categorization, and severity rating of usability problems with the same experimental rigor called for in other parts of the design. One way to reduce instrumentation problems is to have multiple blind raters (people other than the experimenters) categorize and rate problems. In the ideal case, the raters would not have knowledge of either the conditions or participants. In addition, the order in which problems are rated should be randomized or carefully counterbalanced across raters. Measures of inter-rater reliability, such as Cohen's Kappa, also should be computed and reported."

Measures of reliability including Cohen's Kappa and others are considered elsewhere in Section 2.5 (Metrics).

In discussing "Causal Construct Validity", they advise that,

> "The prime responsibility of individual researchers on these issues is to provide explicit information about the exact operations and methods used."

> "If the same participants are exposed to more than one treatment, then the standard operating procedure of experimental design is to counterbalance."

These recommendations are used during the testing of playthrough evaluation, presented later in Chapter 7 (Testing Playthrough Evaluation).

Intrinsic and Payoff Measures of Usability

Gray and Salzman (1998) enumerate a number of potential problems when comparing between usability evaluation methods. The most important of their concerns is with "Effect Construct Validity". Citing Scriven (1967), they use the terminology introduced in Section 2.3 (Evaluation). In particular they make the distinction between "intrinsic" and "payoff" measures of usability. Intrinsic refers to aspects of a design that can be inspected, and payoff refers to how they are used in user test, particularly with respect to performance and other usability qualities. Empirical usability evaluation methods measure payoff, whereas analytical usability evaluation methods infer and predict payoff.

> "Empirical usability evaluation methods can identify problems, but care must be taken to isolate (e.g.,Landauer, 1988) and identify the feature that caused the problem. None of the studies we reviewed report systematic ways of relating payoff problems to intrinsic features; all apparently rely on some form of expert judgment."
>
> "Analytic UEMs examine the intrinsic features of an interface in an attempt to identify those that will affect usability (the payoff) in some way: errors, speed of use, difficulty of learning, and so forth."
>
> "...analytic UEMs must seek to relate intrinsic attributes to usability payoffs"

The novel method presented later in Chapter 6 (The Playthrough Evaluation Framework), playthrough evaluation, reconciles these two sides of evaluation by employing both empirical and analytical phases.

Validity in Problem Matching

Gray and Salzman (1998) note that another threat to validity is in the mapping from individual problem tokens to general problem types or categories. In their terminology, a problem token is an individual problem report. Tokens that are reported, perhaps by different evaluators of the same system, are grouped together into the same problem type or category. The grouping of similar problem reports is a concern that is returned to in Chapter 6 (The Playthrough Evaluation Framework) and Chapter 7 (Testing Playthrough Evaluation). The novel method developed in those chapters, playthrough evaluation, presents an approach to formalise this stage of the evaluation. Gray and Salzman (1998) propose that a common scheme for categorising problems would be beneficial in order to compare problems across individual systems, interfaces, and studies.

> "Developing a common categorization scheme, preferably one grounded in theory, would allow us to compare types of usability problems across different types of software and interfaces"

The user action framework goes some way to address this concern. It is based in theory, has been used in many studies, and defines a wide and detailed taxonomy of usability issues.

However, later in this thesis in Chapter 6 (The Playthrough Evaluation Framework), limitations to the user action framework are described in regards to it's applicability for video game evaluation. An adapted model is derived that still maintains a connection to the original scheme and underlying theory, facilitating some measure of consistency between the two.

Clearer Procedures Ameliorate the Methodology Effect

Playthrough evaluation, introduced later in this thesis, involves a greater degree of specificity in methodology than the more informal approaches described earlier, with more rigorous and clearly defined procedures to be followed. This does not eliminate evaluator subjectivity and expertise, which, after all, is clearly a valuable - if nebulous - commodity. What it does, however, is to isolate and account for those areas where evaluators' personal opinions need to be expressed.

The other aspects of the evaluation procedure can easily be compared to one another, and the decision making process of each evaluator is made explicit and recorded to facilitate comparison and sharing of interpretation. This is a similar approach to that advocated by structured evaluation methodologies such as Matera et al. (2002), where concrete procedures are defined for novices to follow during evaluation.

Structured Methods Ameliorate the Evaluator Effect

A. P. O. S. Vermeeren et al. (2003) describe the structured evaluation of user test footage using the DEtailed Video ANalysis (DEVAN) method. Any-Two was used to calculate agreement between two evaluators' coding of the same video footage. High inter-evaluator reliability rates were reported across several studies, varying from 53% and 80%. However, these high levels can be explained as the evaluators were experts with the method being used as they had authored it, and had used it in studies throughout some years previously. It is reasonable to assume that as they had previously collaborated, they had a considerable degree of implicit understanding of the indicators and method in agreement with one another. Whether these results can be replicated by independent evaluators, particularly novices, seems unlikely.

A. P. O. S. Vermeeren et al. (2002) note several areas in which evaluator subjectivity can affect the results of evaluation:

- Logging: What source data to record.
- Transcription: What and how to transcribe the data (e.g., verbal, non-verbal; codes; etc.)
- Inference: How and when to assert user goals and intentions when they are not directly communicated by the user.
- Problem identification: How to categorise and merge problems. Defining events indicative of usability problems.

They argue that a methodological tool should provide guidance with these questions. Their tool, DEVAN, does not address how to categorise different types of problems, only what objective indicators are available. The user action framework, described later in this thesis, directly deals with interaction breakdown categorisation, and is a useful complement to problem discovery.

2.4.2 Conclusion

Poor Reliability Underpins the Evaluator Effect

A great deal of the evaluator effect is due to unavoidable subjective individual differences in evaluator experience. However, there are many areas of usability evaluation that currently contribute to the effect mainly due to poorly specified processes. This thesis examines how to ameliorate the evaluator effect by improving reliability of usability evaluation primarily through improvements to methodology.

Particularly in the discount methodologies reviewed here, problems of poor reliability are exacerbated due to deficiencies in the methodology itself, rather than being solely qualities of the participant. When a methodology is underspecified, it is understandable that evaluators will conduct their evaluations in somewhat different ways, and hence produce different results.

Hornbæk and Frøkjær (2008) conducted a large scale study exploring the evaluator effect, and comment that

> "...usability evaluators report substantially different sets of usability problems when applying the same evaluation technique on the same application."

They note in particular that methodologies without clear guidelines contribute to the evaluator effect,

> "...vague evaluation procedures may make evaluators focus on different things during the evaluation"

The way to resolve this is to define more rigorous and structured evaluation procedures. Clearer guidelines and explicitly defined criteria leave less possibility for ambiguity, hence increasing the repeatability and validity of the evaluation. While there will always be the possibility for interpretation and disagreement, in the case of using well defined processes the effect is not attributable primarily to the methodology itself. Furthermore, by controlling for this, it is possible to explore the remaining evaluator effect. For example, given a reliable and repeatable methodology, intra-rater variance can be examined for learning effects. Similarly, evaluator ability can be quantified against known benchmarks.

Inspectability of Procedures Helps Improve Methods

Perhaps most importantly, by explicitly exposing the procedures for evaluation, it becomes possible to evaluate the methodology itself. Previously, with opaque evaluation procedures which involved a great deal of subjective opinion, and unreported private group negotiation to resolve the inevitable differences, it is very difficult to identify areas for improvement. This intention is in line with E. L.-C. Law and Hvannberg (2004b),

> "We aim not only to improve the content (i.e., heuristics) but also the method of HE."

A. P. Vermeeren et al. (2008) explores the concept of inspectability in a comparative study conducted by three independent usability labs evaluating the same product using the same

standardised method. Despite controlling the data and process well they still observed poor consistency of evaluation results. They acknowledge that absolute objectivity is not possible, as the evaluator is an inherent part of the evaluation. Particularly given that objectivity cannot be guaranteed by a method alone, they argue in favour of evaluations being open for inspection, thus allowing confirmation or falsification. They define the criterion of inspectability as requiring that,

> "...both the original data and the processes used to compress these data should be available to be inspected and confirmed by outside reviewers of the study."

Publishing evaluation results for others to examine and compare against is a very useful approach. This ideology is used later on, in Chapter 6 (The Playthrough Evaluation Framework), where a novel methodology is proposed, playthrough evaluation.

This is predicated on the idea that a clear methodology can be published alongside test data that others can evaluate. Using the same test data, and the same clear evaluation methodology, the same results should be produced. If they are not, then it is possible to identify where the disagreements took place, to investigate why they occurred, and to remedy the methodology such that they do not occur for subsequent evaluations. These corrections should be published so that the evaluation knowledge uncovered can be shared by the rest of the research community. This approach relies on the idea that the evaluator effect can be managed by a more clearly defined methodology.

2.5 Metrics

The various approaches to quantifying usability evaluation are reviewed in this section. A particular emphasis is given to reliability, and the role it plays in validation, especially during the problem discovery and analysis stages of evaluation.

The research questions posed in this section include:

- How are usability evaluations quantitatively compared?
- What metrics are available for usability evaluation?
- What procedures are required for metric computation?

Usability is commonly interpreted with metrics such as completion rate and errors for Effectiveness, and time on task for Efficiency (Sauro and Kindlund, 2005). Satisfaction is a more complex outcome, particularly for video games as aesthetic experiences, where player preference may be a significant component, where tasks are intentionally challenging, and where failure does not necessarily result in a negative player experience.

2.5.1 Thoroughness, Validity, Reliability, and Effectiveness

Bastien and Scapin (1995) defines three key terms that subsequently inform the literature,

- Thoroughness is an attempt to address the "widest scope of the interface".
- Validity is intended to "evaluate systems on those aspects the dimensions are intended to evaluate".
- Reliability tests for the "same results under the same conditions".

The following definitions are also provided by Sears (1997),

- "Thoroughness measures the percentage of the problems that are being found."
- "Validity measures how much extra effort is being spent on issues that are not important."
- "Reliability provides a measure of the consistency among different evaluations."

An additional metric is sometimes reported in the literature, synthesised as the product of Thoroughness and Validity. Note that this is unrelated to the usability performance metric effectiveness, as in the traditional three core usability aspects: efficiency, effectiveness, and satisfaction.

Also provided are formulae for quantifying the thoroughness, validity, and reliability of usability inspection methods, which are described in the following sections.

2.5.1.1 Thoroughness

Thoroughness is the ratio of real problems that are identified to the number of problems that actually exist:

$$\text{Thoroughness} = \frac{\text{Number of real problems identified}}{\text{Number of real problems that exist}} \quad \text{(eq. 1)}$$

The range of values for this metric is from 0 to 1 with the following meanings,
1.0 = The usability inspection method predicts all of the actual problems in the system.
0.0 = The usability inspection method does not predict any of the actual problems in the system.

The number of real problems that exist can be difficult to ascertain. One proposed approach is to evaluate a given system, then intentionally add new problems to it. This is clearly difficult to perform, particularly for summative cases. More often a value is simply estimated by user testing. The identification of *potential* problems is usually the result of another usability inspection method such as heuristic evaluation.

2.5.1.2 Validity

Validity can be measured as the ratio of "real" usability problems identified to all issues identified as usability problems,

$$\text{Validity} = \frac{\text{Number of real problems identified}}{\text{Number of (predicted) problems}} \quad \text{(eq. 2)}$$

The range of values has the following meanings,
1.0 = The usability inspection method does not make predictions that do not appear in user testing.
0.0. = The usability inspection method only predicted problems which were not reported during user testing.

2.5.1.3 Reliability

Two metrics commonly used to determine reliability are Cohen's Kappa and the Any-Two agreement. As noted in Barendregt (2006), Cohen's Kappa is only applicable for use with 2 evaluators who rate the same set of data. Using freeform problem detection, evaluators' sets of observed problems usually differ to one another, and so Any-Two agreement is more commonly used (Barendregt, 2006; Hertzum and Jacobsen, 2001).

Kappa With Pre-Determined Lists of Issues

Cohen's Kappa is a statistical measure of agreement between two raters, for cases where a fixed list of issues is known in advance.

Statistical significance for kappa is rarely reported, probably because even relatively low values of kappa can nonetheless be significantly different from zero but not of sufficient magnitude to satisfy investigators. What's more, as the number of codes used in the rating increases, kappas become higher. Interpretations for values of Cohen's Kappa are suggested by Landis and Koch (1977):

- < 0 indicate no agreement.
- 0-0.20 slight.
- 0.21-0.40 fair.
- 0.41-0.60 moderate.
- 0.61-0.80 substantial.
- 0.81-1 almost perfect agreement.

Fleiss (1971) defines kappa for measuring agreement of nominal data when using two or more coders. Similar interpretations for values apply as with Cohen's Kappa. It can be interpreted as expressing the extent to which the observed amount of agreement among raters exceeds what would be expected if all raters made their ratings completely randomly.

Any-Two

Any-Two is used throughout the literature to compare the reliability of evaluations (Barendregt and Bekker, 2006; Barendregt et al., 2006; Capra, 2006; Frøkjær and Hornbæk, 2008; Hertzum and Jacobsen, 2001, 2003; Hornbæk and Frøkjaer, 2008; Hornbæk and Frøkjær, 2008; Hvannberg et al., 2007; Lanzilotti et al., 2011; E. L.-C. Law and Hvannberg, 2004a; A. P. O. S. Vermeeren et al., 2003; A. P. Vermeeren et al., 2008).

It is a measure of agreement between two or more raters, expressed as a percentage from 0% for absolutely no agreement to 100% for complete agreement between all evaluators. Hertzum and Jacobsen (2001) define it as,

> "...the number of problems two evaluators have in common divided by the number of problems they collectively detect, averaged over all possible pairs of two evaluators."

That is,

$$\text{Any-Two Reliability= mean of } \frac{|P_i \cap P_j|}{|P_i \cup P_j|} \text{ over all } {}^1\!/_2\, n(n-1) \text{ pairs of evaluators.} \qquad \text{(eq. 3)}$$

Where P_i and P_j are the sets of problems identified by evaluator i and j, and n is the number of evaluators.

It should be noted that this is not a linear scale, and that Any-Two values tend to be smaller than Cohen's Kappa. For example, consider if two evaluators, e_1 and e_2 respectively found problems (p_a, p_b), and (p_a, p_c). In this case they each discover two issues, where one of their two (p_a) was also discovered by the other evaluator. Sharing 50% of the issues with another evaluator sounds like good reliability, but Any-Two would be computed as:

$|P_i \cap P_j| = 1$
$|P_i \cup P_j| = 3$
i.e., $\frac{1}{3}$

Hertzum and Jacobsen (2001) reviewed 11 publications in the literature that addressed the evaluator effect. They argue against the use of Kappas like Cohen's Kappa and Fleiss as they assume that the number of problems in the system is known in advance or estimated to a reasonable degree of certainty. They also assert that the number of evaluators used are too low in the publications reviewed by their paper.

2.5.1.4 Effectiveness

Sears (1997) defines Effectiveness as simply the product of two other metrics,

$$\text{Effectiveness= Thoroughness x Validity} \qquad \text{(eq. 4)}$$

Following Hartson et al. (2001), E. L.-C. Law and Hvannberg (2004b) however consider that a raw count of hits does not constitute a real indicator of effectiveness, and that "actual effectiveness" be computed by taking severity into account. Furthermore, they build on the differentiation of "superficial" and "actual" effectiveness from Cockton, Lavery, and Woolrych (2003); Cockton and Woolrych (2001),

- General (superficial) Effectiveness:
  Number of hits.
- Specific (actual) Effectiveness:
  Hits categorised by frequency, severity, and discoverability.

$$\text{Actual Effectiveness= Thoroughness x Validity} \qquad \text{(eq. 5)}$$

Where Thoroughness and Validity are computed with respect to problems detected in user testing (UT-UP) and heuristic evaluation (HE-UP),

$$\text{Thoroughness=} \frac{\text{Sum of severity ratings of all Hits}}{\text{Sum of severity ratings of all UT-UPs}} \qquad \text{(eq. 6)}$$

$$\text{Validity=} \frac{\text{Number of Hits}}{\text{Total number of HE-UPs}} \qquad \text{(eq. 7)}$$

In addition they also propose a novel formula for computing efficiency of the usability evaluation method,

$$\text{Actual Efficiency=} \frac{\text{Number of Hits identified}}{\text{Number of Hours invested}} \qquad \text{(eq. 8)}$$

2.5.2 Metrics Through the Lifecycle

Thoroughness and validity are important in that they determine how useful or relevant the results of an evaluation will be for helping to improve a product. These factors are especially important during formative testing. The purpose of testing in that stage of a product's lifecycle is to identify problems with early prototypes that could have an impact on the user's experience of the final product. Reliability is of course also an important consideration during formative testing; ideally all evaluators should be able to come to the same conclusions regarding the problems that a system might have even when only evaluating a prototype. However, the importance of reliable evaluation is especially increased during summative evaluation. When a product has been completed to a sufficient state to be tested in this way, it should be possible to unambiguously identify and define specific problems with the actual system using real users. More informal approaches to evaluation could be used during the formative stage where a system is still in development. At this stage there is greater scope for individual, subjective interpretation, and creative ideation may be more important than analytical rigour, as implied by Scriven (1967).

A more detailed discussion around the distinctions between these two forms of evaluation is presented in Section 2.3 (Evaluation)

2.5.3 Metrics Require Procedures

Defining metrics for usability evaluation method comparison holds some promise for quantifiably comparing between which is better. However, caution is needed. Most of the studies reported in this literature review lack well defined procedures. Often there are no formal definitions for what constitutes a usability problem, or how to determine when separate evaluators have identified the same problem. These questions undermine the metrics used to compare between usability evaluation methods. When there are gaps in the procedures that produce data used by the metrics, then the metrics themselves lack validity. Gray and Salzman (1998) take care to discuss issues of validity in evaluation practice and present a comprehensive analysis of where these problems can occur. They recommend that more rigour is applied to the process. The novel framework defined later in this thesis, the player action framework, is designed to address these concerns. It defines procedures that are absent from traditional usability evaluation methods, and shows in Chapter 7 (Testing Playthrough Evaluation) how metrics can be computed within this framework.

This section reviewed the metrics used by usability evaluation methods and also used in the comparison between usability evaluation methods. One of the key points made in this section is that caution is needed when trying to treat metrics independently of the methods they are used with.

The following proposals are made for the studies presented later in this thesis:

- Use Any-Two to compute reliability.
- Define clear procedures for problem discovery and analysis, particularly problem matching, so that reliability can be computed for each stage..

2.6 Heuristic Evaluation

Heuristic evaluation has more than any other method been widely used to evaluate video games. Many separate sets of heuristics have been developed to address different styles of game, platforms, and qualities of usability and player experience. Given its development and use in the research community the literature for this method is explored in more detail in this chapter, and becomes a focal point for the remainder of this thesis.

The available literature on heuristic evaluation was first informally reviewed in order to get an impression of its use in the research community. During the review it became clear that many publications used heuristic evaluation in different, unspecified, or informal ways. The literature was then reviewed again, and notes were made on how the method was used in each publication.

The research questions asked in this section are:

- What is heuristic evaluation?
- What are the strengths and weakness of this method?
- How is heuristic evaluation applied?
- How are heuristics validated?
- What heuristics are appropriate for video game evaluation?

The results of this section are used throughout the studies later in this thesis, and in particular in Chapter 4 (Testing Heuristic Evaluation for Video Games).

2.6.1 Background

Heuristic Evaluation is a Discount Method

> "The discount usability engineering philosophy aims at increasing the use of usability methods by reducing their perceived cost and complexity." Nielsen (1994b)

Major advantages of heuristic evaluation in particular were originally cited by Nielsen and Molich (1990) as

- "It is cheap."
- "It is intuitive and it is easy to motivate people to do it."
- "It does not require advance planning."
- "It can be used early in the development process."

Discount Methods Are Well Suited to Formative Evaluation

These qualities describe the benefits of a "discount" method that's particularly well suited for use in formative stages of product development. However, the method has subsequently been widely used for summative evaluation too. This is problematic in light of the requirements of summative methods to be more rigorous than formative methods (Scriven, 1967), as heuristic

evaluation suffers from problems with inter-evaluator reliability. Nonetheless such use of the method is commonplace, particularly for video game evaluation, possibly because there are no other methods that are so well developed specifically for this domain.

2.6.2 Application

As described earlier in Section 2.3 (Evaluation), the main stages involved in evaluation are implicitly problem discovery and problem analysis. These stages of heuristic evaluation are summarised in the following sections.

2.6.2.1 Problem Discovery

Common practices of problem detection typically come in two forms:

- Free-form, system-focussed
- Prospective, heuristic-focussed

In the free-form and system-focussed form the evaluator attempts to identify issues using their expert opinion. In the prospective and heuristic-focussed form the evaluator considers each heuristic and examines the ways in which the game conforms or violates it. Problems found using free-form discovery are sometimes then retrospectively matched to one or more heuristics.

Inter-evaluator reliability can be computed to confirm whether evaluators identify the same issues, and categorised with the same heuristics.

2.6.2.2 Problem Analysis

In free-form detection, heuristics can be used post-hoc to retrospectively categorise issues. This allows the option of assigning zero or more heuristics to describe each issue. Inter-evaluator reliability is likely to be lower in this case. In feed-forward detection, each issue will usually be assigned to exactly one heuristic. Inter-evaluator reliability can be measured and is likely to be greater than in the retrospective form. After each individual evaluator has conducted their own evaluation, it is useful for all evaluators who are evaluating the same product to compare and merge their results.

Andre et al. (2001) comments that the lack of a structured framework in heuristic evaluation means that it is difficult to compare between different between usability problems. In particular this is due to a deficiency in the heuristics to identify specific qualities that would constitute a problem. Poor reliability is exacerbated further as multiple heuristics can be interpreted as addressing the same issues. This confusion is addressed by Cockton and Lavery (1999) who discuss how some heuristics address potential problems without specifying causes, as well as the opposite condition, a cause that does not necessarily imply a problem.

2.6.3 Validation

Many Heuristics Are Unvalidated

In a recent review of empirical user experience research (Bargas-Avila and Hornbæk, 2011), it was shown that many empirical studies use methodologies without validation. This is particularly troubling as errors in the validation process can affect any final validity results. However, when heuristic evaluation is validated it is usually performed by comparing against the results of user testing.

Validating Against User Tests

In order to validate a heuristic evaluation, it is typical to compare the results to those of a user test.

Validating heuristic evaluation is principally concerned with the problem discovery, where a comparison is made between the problems discovered by heuristic evaluation and those discovered during user testing. The comparison can include whether both methods discovered issues in the same part of the system, as well as whether the same issues were analysed in the same way.

In some cases, heuristic evaluation is compared to user testing only using a pre-determined set of problems. That is to say that only the analysis stage of heuristic evaluation is validated, not the initial problem detection stage that normally then leads into analysis. For example, Andre et al. (2001) describes a study in which 10 usability professionals matched 15 predefined usability reports each to a single one of Nielsen's ten heuristics (Nielsen, 1994a). Problem detection itself was not tested, as the analysts started with a pre-determined list of issues.

Problem discovery in heuristic evaluation is usually a predictive process. Evaluators consider the system and make predictions about which issues are likely to cause users difficulties. Validating these predictions is performed by comparing the *potential* issues discovered by heuristic evaluation to the *actual* issues discovered by user testing. Minimally this would consider whether issues were discovered in the same parts of the system, but could also include whether the issues are analysed in the same way too.

Both of the stages of heuristic application and validation are examined in much more detail in the following sections. Heuristic evaluations from the literature are reviewed, and each stage is examined in detail. The principle points considered are,

- Application:
  How are candidate issues detected?
  How are problems classified?
  How are problem tokens merged, and filtered?
  What metrics and processes are used to quantify the reliability of each stage?
- Validation:
  In user testing, how are candidate issues detected?
  How are issues classified?
  How are problem tokens merged, and filtered?
  What metrics and processes are used to quantify the reliability of each stage?
  How are issues that were detected in user testing and heuristic evaluation compared to one another?

2.6.4 Application

This section reviews the approaches described in the literature for how heuristic evaluation is conducted. In particular the following stages are examined:

- Problem discovery.
- Problem analysis, especially comparison between discovered problems.

Each of these stages in the application of heuristic evaluation introduces the possibilities for error, which are compounded in later stages.

The following research questions are asked,

- How are issues discovered in heuristic evaluation?
- How are issues analysed in heuristic evaluation?
- How are issues compared in heuristic evaluation?
- What metrics are used to quantify the comparisons?
- How valid and reliable are the comparisons?
- What stage in the product's lifecycle is the usability evaluation method used?

Examples from the literature are considered, with particular attention for how these stages are addressed. The section concludes with a summary and critique of current practice, and recommendations for improvements.

2.6.4.1 Problem Discovery

As a discount method, heuristic evaluation lacks a well defined formal procedure, and many applications of it use slightly different approaches.

In order to help explicate why and how errors are introduced in heuristic evaluation, novel terminology is proposed to distinguish between two distinct forms of problem detection used in the literature,

- Feed-forward (**prospective**).
- Post-hoc (**retrospective**).

Prospective heuristic evaluation is heuristic-focussed, where evaluators start with the heuristics and look for violation or conformity of them in the system. In contrast, *retrospective* evaluation is typically more free-form. In this mode evaluators begin by looking for issues in the system, then once they have been discovered they are then categorised against the set of heuristics afterwards.

2.6.4.2 Prospective

In a feed-forward or prospective approach (such as Desurvire et al. (2004), Pinelle et al. (2008a)) each heuristic is examined, and the system considered for ways that it conforms or violates them. This approach functions as both problem detection and problem classification. Prospective evaluation may result in errors of omission, where some real issues are not noticed if none of the heuristic descriptions prompt the evaluator to look for them. In addition, errors of commission may also occur where the heuristics encourage evaluators to detect potential issues that do not actually cause problems in real use (false positives).

2.6.4.3 Retrospective

The alternative is to use heuristics in a post-hoc, or retrospective manner. With this approach, once issues have been identified (perhaps using a freeform or system-scanning procedure) they are then matched to the heuristics that best explain them. This approach is often used, such as in Korhonen et al. (2009). This is not so much a problem detection approach, as problem classification. As such a potential drawback is that the approach relies on the evaluator's informal ability to detect issues. This is particularly problematic with the most common form of freeform evaluation, as evaluators are likely to overlook many issues that do cause problems for real users. Structured approaches such as system-scanning might help to ameliorate this.

This terminology is used in the remainder of this literature review to classify the approaches taken by the research community.

Heuristic Evaluation Provides Limited Discovery Resources

In general, most of the examples in the literature employ an informal and unguided approach to problem discovery. In the terminology from Cockton et al. (2004) this would be described as "System Scanning".

Cockton and Woolrych (2001) also reflected on the usability evaluation method in general, and concluded that the lack of a structured approach limits its effectiveness, even going so far as to assert that,

> "...heuristics do not support the discovery of possible usability problems."

This also has implications for problem analysis, in that only a subset of the issues present in the system are likely to be detected using this approach. Analysis conclusions based on this subset will necessarily suffer from reduced inter-evaluator reliability as the source data discovered by each evaluator is likely to be different to one another, primarily due to errors of omission in the problem discovery stage. The danger that this presents is that any conclusions drawn from such studies may suffer from reduced validity,

> "...heuristics are used inappropriately and, if acted on, would result in poor problem remediation."
>
> Cockton and Woolrych (2001)

Later in this thesis, a novel method, playthrough evaluation, is defined with a more structured, method following approach to address this.

2.6.4.4 Problem Analysis

Comparing Problems

Once evaluators have individually compiled sets of problems, they are then usually compared to each other and merged into a master list. This stage is variously called problem matching (Hornbæk and Frøkjær, 2008), duplicate filtering/reduction (Connell and Hammond, 1999), merging (Cockton and Woolrych, 2001), aggregating (Sim and Read, 2010), or consolidating (E. L.-C. Law and Hvannberg, 2004b, 2008; E. L.-c. Law and Hvannberg, 2008).

Generally speaking, if multiple evaluators identify the same problem using the same heuristic, then the individual problem reports are said to match and are merged into a single unique problem report. Usually there is no formal method for determining whether individual problem reports are matched, though. Master lists are usually produced through collaboration and discussion with multiple judges. Sometimes judges work independently to produce their own personal master lists, which they then discuss and merge afterwards. Alternatively multiple judges can work collaboratively to develop a single master list together. The former approach is preferable as it allows for inter-evaluator reliability metrics to be computed.

Informality in Traditional Domains

When comparing seven independent evaluations of the same system, Kotval et al. (2007) report an overlap in the problems detected by each evaluator of just 14%. No formal procedure is reported for how the overlap was detected, however. They conclude by pointing out possible problems in method and scope that may have contributed to these results,

> "Low overlap may indicate inconsistency of the evaluation criteria used by individual evaluators or incompleteness in the coverage of all usability problems"

A further example is given by Sim (2009) which describes several evaluations where lists of issues discovered by independent evaluators were then informally merged, and filtered into a single super list. No figures for inter-evaluator reliability were computed. Another study in the same publication describes how individual problem lists from independent evaluators were merged in an open card sort by 4 other researchers. 2 analysts then informally compared user test lists to the heuristic evaluation lists. Although cases of disagreement were mentioned, no figures were presented, and no reliability data were computed.

Similarly, Andre (2000) reported a formative evaluation of a message management system consisting of both web and voice interfaces. 3 novice evaluators conducted a scenario-based heuristic evaluation using retrospective heuristic assignment to a single heuristic. The three evaluation reports were informally compared and filtered, but no data for reliability were computed.

Informality in Summative Game Evaluations

Many summative evaluations also exhibit rather informal processes. This runs counter to the recommendations of Scriven (1967) which advises formal procedures and criteria for evaluation. Examples are described in the following.

In Korhonen (2010), 2 experts conducted a summative expert evaluation of a beta version of a commercially available mobile game. Similarly to the other studies reported in this literature review, no inter-evaluator reliability data were computed. A think aloud user test was then conducted with 6 players. 2 observers informally noted down any observations of issues. After the test session the issues were informally examined and merged together. Each issue was then assigned to a single heuristic, and the sets of user test and heuristic evaluation data were then compared to one another. No formal procedure was used so it would not be possible to validate this approach.

5 novices conducted a summative evaluation of a PC RTS game in Pinelle et al. (2008a). The 3 authors of the paper considered the evaluation reports, and informally discarded items they considered to be duplicates, or not usability problems. They informally compared the 5 sets of reports, and felt that multiple evaluators had reported the same issues, though no quantitative analysis was conducted, and no inter-evaluator reliability or other data were recorded. None of the standard metrics on reliability were computed for problem detection, categorisation or severity assignment. Furthermore, as no user testing was conducted, no data for thoroughness, validity or effectiveness could be provided either.

Barendregt et al. (2003) presents a summative evaluation of a children's video game, though they discuss that their method could be applicable for formative evaluation too. 4 evaluators reviewed video footage of 4 user test play sessions and assigned heuristics to describe the issues they discovered. All of the evaluators' reports were merged, but no data or procedure were reported for how the merging was conducted, so no claims can be made about the validity of the data.

Korhonen et al. (2009) describe an experiment where 4 teams of 2 evaluators conducted a summative heuristic evaluation on a mobile game. After the evaluations, the issues were informally compared across the teams. Some were discarded, and some were identified as addressing the same problem, though no procedure was described for how this took place.

Informality in Formative Game Evaluations

These less formal procedures are especially common in formative evaluations, though less problematic, such as seen in the following.

One evaluator conducted a formative heuristic evaluation on a game in Desurvire et al. (2004). As only a single evaluator was used, no data on the reliability of the heuristics is possible.

Desurvire and Wiberg (2010) conducted a formative user test using four unspecified, unfinished console games. Each of the issues identified were assigned to a single heuristic. After the play session, the evaluator subjectively decided which issues did not deal with learning and having fun, and rejected these from further analysis. The remaining issues were categorised as either accessibility / approachability, or usability / playability, though no definitions for these terms are provided, nor are the procedures described for how the categorisation of each issue took place. As only a single evaluator was used in the user test, the data cannot be considered for reliability. Also, as the issues identified in user test were filtered, but the issues in heuristic evaluation were not, any conclusions drawn from this data have limited validity.

2.6.4.5 Summary

A number of potential problems have been identified in the practice of heuristic evaluation reviewed in the literature. Principally the lack of formality calls the studies' validity into question. Many studies did not compute metrics for reliability, but rather resolved disagreements through private discussion. While their results may have been useful for their own individual projects, the lack of clarity in their process prevents them from being critiqued and improved, or even reproduced.

Proposals for improvements to the evaluation methodology are summarised as follows,

- Problem discovery.
  Follow a systematic and formally defined approach so that it is possible for independent evaluators to reproduce the process and validate the results of one another.
- Problem analysis.
  Most analysis is currently cursory, and usually relies on an evaluator choosing a single, relatively broad heuristic to describe the issue. A structured or formal problem description would facilitate the comparison of issue reports between evaluators, and enable independent researchers to reproduce and validate published results.

2.6.5 Validation

This section considers the process of matching problem predicted during heuristic evaluation with those observed during user testing. This is sometimes performed based on purely textual descriptions without reference to the original source data (e.g., video footage).

Current approaches are considered and potential problems are identified that risk the validity of results. This review particularly focusses on the stages of problem discovery and analysis. Examples from the literature are introduced, and the approaches used in these stages are described and critiqued.

This section concludes with proposals for how to overcome these problems so that heuristic evaluation can be validated appropriately.

Research questions for this section address problem discovery and analysis in user testing,

- How are issues discovered in user testing?
- How are issues analysed in user testing?
- How are issues matched in user testing?

Problem discovery and analysis in heuristic evaluation,

- How are issues discovered in heuristic evaluation?
- How are issues analysed in heuristic evaluation?
- How are issues matched in heuristic evaluation?

And how they are compared to one another,

- How are comparisons made between user test and heuristic evaluation data?
- What metrics are used to quantify the comparisons?
- How valid and reliable are the comparisons?

Differences Between Inspection and Empirical Methods

Comparison of issues predicted by heuristic evaluation data against empirically observed user test data is an important stage of validation, as empirical user testing is sometimes considered to be the gold-standard of usability evaluation methodologies. However, this comparison is not trivial, as each method has a different application, purpose, and results.

Doubleday et al. (1997) conducted an experiment comparing heuristic evaluation and user testing, but found differences in the kind of conclusions reported in each condition,

> "Heuristic evaluators were, naturally, observing the interface and were not absorbed in using the system to perform a task. In contrast, end users were visibly absorbed, using the system to perform specific tasks. The difference in emphasis is one of the reasons why the heuristic evaluators failed to identify some of the end users' task based problems."
> Doubleday et al. (1997)

Cockton and Lavery (1999) suggest that matching of predicted and actual observed user test issues should be considered as a spectrum, not necessarily as a binary decision. This kind of nuanced approach will be especially important in cases where the terminology and report format used in user test evaluations and expert predictions differ, which is the case in most situations. However, this introduces further ambiguity in the comparison between usability evaluation methods. Comparisons would be open to more subjective interpretation and disagreement between analysts. Another approach would be to employ the same terminology and reporting format used by both usability evaluation methods, to make it clearer when heuristic evaluation data and user test data addressed the same issue. The novel methodology, playthrough evaluation, presented later in Chapter 6 (The Playthrough Evaluation Framework), addresses this problem by maintaining a consistent form for transcription, evaluation, and prediction throughout the methodology. This consistency allows greater confidence in the matching process between issues, whether they are actually observed in user testing, or predicted to occur through inspection with heuristic evaluation.

Informal Validation in Traditional Domains

Doubleday et al. (1997) conducted an experiment to explore the differences between heuristic evaluation and user testing when evaluating a database visual interface. However, no procedure was reported for how issues were discovered or recorded, nor for whether any problem duplication / merging / matching took place.

Sim (2009) considered a validation of Nielsen's heuristics against a user test and subsequent questionnaire. One evaluator informally decided which responses addressed usability issues. Two evaluators informally merged and coded the data for which of the fixed set of tasks the users were attempting. These were informally merged but no inter-rater reliability was reported.

The heuristic evaluation data were collected by 11 evaluators who informally identified usability problems in a freeform, ad-hoc, unstructured manner. 4 analysts then conducted an open card sort together to informally derive new categories from the issues. When analysts informally agreed that more than one evaluator had identified the same issue, they were merged

into a single category. As this process was conducted informally and as a group, no inter-evaluator reliability data were reported.

Informal Validation for Game Heuristics

Korhonen (2010) does not derive heuristics, but rather starts with a pre-made set. The purpose of that paper was to compare the effectiveness of heuristic evaluation with playtesting, employing the same set of heuristics for both. The paper concludes that the expert review method was able to identify problems as accurately as playtesting, though none of the standard metrics for effectiveness, reliability, thoroughness or validity were provided. Furthermore, several stages in the comparison rely upon subjective interpretations in earlier stages.

The 10 heuristics derived from video game reviews in Pinelle et al. (2008a) were validated against heuristic evaluation only, without any user testing. However, none of the standard metrics on reliability were computed, and as no user testing was conducted, no data for thoroughness, validity or effectiveness could be provided.

Desurvire et al. (2004) had 4 participants play an unspecified game prototype, which consisted of navigable screens but no gameplay. The evaluator used an unspecified coding scheme to identify and record player actions, comments, failures and missteps. Each issue was assigned to a single heuristic, but as only a single evaluator was used, coding and severity data were not considered for reliability.

Desurvire and Wiberg (2010) conducted a user test using four unspecified, unfinished console games that they had earlier used for heuristic evaluation. One evaluator observed and coded player behaviour, then after the play session the evaluator subjectively decided which issues did not deal with learning and having fun, and rejected these from further analysis. The remaining issues were categorised as either accessibility / approachability, or usability / playability, though no definitions for these terms were provided, nor were the procedures described for how the categorisation of each issue took place. As only a single evaluator was used in the user test, the data were not considered for reliability.

2.6.5.1 Summary

Heuristic evaluation is generally compared to user testing. However, this validation typically lacks formality and many of the stages of problem discovery and analysis are glossed over. As the procedures are undocumented, reliability data are usually not reported. In order to ensure that this kind of comparison is a valid one, more rigour is needed particularly in the analysis stages of comparing problems between the two different usability evaluation methods.

The key questions for a reliable validation are,

- Discovery:
  Do users discover the same issues as one another in the user test condition?
  Do evaluators discover the same issues as one another in the heuristic evaluation condition?

- Analysis:
  Do evaluators analyse the issues in the same way as one another in the heuristic evaluation condition?

> Given the same set of user test data, do evaluators analyse the same issues with the same heuristics?

This review of validation approaches in the literature suggest several gaps that raise doubt about the process of validation. These can be addressed by defining clearer procedures to follow for both user testing and heuristic evaluation.

Proposals are as follows:

- Define an annotation scheme for the description of user test data.
  This would allow analysts to document the users' interaction in a standard format which would facilitate clear computation of inter-rater reliability within the user test condition.
- Use the same common annotation scheme to describe potential or actual interaction issues discovered in both user testing and heuristic evaluation.
  This would facilitate clear comparison between the methods, and computation of inter-rater reliability to validate the heuristic evaluation.

2.6.6 Problems With Heuristic Evaluation

Nielsen's heuristics are the canonical material in the literature, though research has suggested that they are deficient for a number of reasons. For example, even when applied to traditional, non-game domains, there is evidence to suggest that they are too general to be applied usefully and reliably (Rohn et al., 2002; Sim, 2009).

2.6.6.1 Lack of Structure Harms Validity

E. L.-C. Law and Hvannberg (2004b) describe a study comparing the effectiveness of Nielsen's heuristics to Gerhart-Powals cognitive engineering principles (Gerhardt-Powals, 1996) but found only small differences between the two sets. However, several major problems were identified with heuristic evaluation generally.

> [an] "unstructured or unsupported approach may undermine the effectiveness of HE."

It was hypothesised that more structured approaches could improve the results.

Lack of Standards Harms the Validity of Comparisons

There is no standard procedure for conducting heuristic evaluations for games. This presents a problem when trying to compare between evaluations, and especially when using different sets of heuristics or evaluators. The lack of standard procedures is also related to the absence of valid and reliable data to conduct a comparison.

Where processes are lacking complete description, any attempt to apply the process again following only the description available will likely result in different outcomes. For example, most heuristic evaluations involve stages where evaluators make subjective decisions, which are then discussed together in groups. Differences of opinion are resolved, but the reasons for the differences, and how they were resolved, are not described. The consequence of this is a process where most of the value is in the private discussions, not in the published

documentation. Procedures, criteria, and insights saved in analysts' minds may be useful for that particular team, but at best may be misleading for other researchers.

Heuristics Lack Specificity

Grudin (1989) points out that abstract design principles such as "Strive for consistency" lack a useful definition with which to identify good and bad examples. He goes on to identify three different types of consistency:

- Internal consistency of an interface design.
- External consistency of interface features with features of other interfaces familiar to the users.
- Correspondence of interface features to familiar features of the world beyond computing.

Furthermore he then shows that these components can be in conflict with one another. Korhonen et al. (2009) reflect on their own heuristics as well as Desurvire's with a similar argument about inappropriate abstraction level.

Doubleday et al. (1997) conducted an experiment comparing heuristic evaluation and user testing, and reported problems with the validity and reliability of heuristic evaluation, stating,

> "Heuristic evaluation problems are often not distinct."

That is to say that they lack specificity, and so could be used to describe a variety of problems, depending on the particular evaluator's interpretation. The lack of specificity is also related to a lack of orthogonality, which would ensure that each heuristic has a clearly separable purpose or area to address. What's more, as there is no standard way to derive and present heuristics there is a great deal of variance between them, even within a single more-or-less coherent set,

> "Evenness - The heuristics themselves are very varied in level and precision. Some heuristics are simple and precise (e.g. Provide feedback and Help/documentation), whereas others are imprecise and difficult to check for completeness (e.g. Prevent errors)."
> Doubleday et al. (1997)

2.6.7 Informality in Evaluation Produces Poor Validity in Report Comparison

Matera et al. (2002) comment that issue reporting in heuristic evaluation is problematic as evaluators do not use a precisely defined terminology. It is argued that a more formal model definition enables structured problem detection and categorisation, as well as reporting format. They argue that using the same terminology for the systematic detection and reporting of problems means an improvement in the precision and comparability of reported issues.

Analytical and Inspection Methods Provide Different Resources

Doubleday et al. (1997) gave a critique of heuristic evaluation and user testing, and noted the difference in types of resources that were available in the two different usability evaluation methods. Echoing the comments of Scriven (1967) regarding the difference between payoff and intrinsic evaluation, they make the point that user testing is good for identifying the outcome of an issue, but perhaps not so good for identifying the cause,

> "...observation alone gives little information as to the cause of the problem, it deals primarily with the symptom. Not understanding the underlying cause has implications for re-design as a new design may remove the original symptom, but if the underlying cause remains, a different symptom may be triggered."

This is particularly important when considering possible solutions. The same study goes on to discuss why reports need detailed and valid conclusions, particularly regarding the causes of usability issues,

> "The complexity of fixing an error will depend on how accurately it has been reported, how thoroughly the cause has been understood"

The argument is made that the causes of issues are more likely to be understood in analytical approaches such as expert inspection and heuristic evaluation. The paper does go on to point out a problem with these approaches as well, arguing that although observation alone is problematic, equally heuristic evaluation alone has deficiencies,

> "...observation of novices is still vital as many problems are a consequence of the users' knowledge, or lack of it, when interacting with a system. Heuristic evaluators cannot place themselves in users' shoes, hence they miss errors."

To address this, Chapter 6 (The Playthrough Evaluation Framework) proposes an adaptation to the heuristic evaluation method that requires evaluators to initially play the game in a natural way as a player. Footage of the playthrough is recorded for later analysis. This initial playthrough gives the evaluator the firsthand experience of the game system itself and the actual player experience. Following this, the evaluator then switches mode from the perspective of player to that of evaluator, and conducts the evaluation in a more critical, reflective way. The evaluation is conducted on the footage of the evaluator's earlier playthrough.

2.6.8 Summary

This section addressed the following research questions,

1. What is heuristic evaluation?
2. What are the strengths and weakness of this method?
3. How and when can heuristic evaluation be applied
4. How are heuristics validated?
5. What heuristics are available for video game evaluation?

The main points made in this section are summarised as follows.

Heuristic evaluation is a usability evaluation method that was popularised by Nielsen which was intended as a formative usability evaluation method, for use with paper prototypes of simple text and telephony systems. Since then it has been used for more complex systems, and as a summative evaluation method, particularly with video games. It is beneficial as a discount evaluation method being quick to use and requiring no real user testing. The principle drawback is that results can be unreliable, as different evaluators have different interpretations of the same system, and can produce contradictory reports.

Heuristic evaluation is usually applied in a freeform manner, where evaluators are free to use a system and identify problems without structured guidance. This lack of a formal procedure is one source of unreliability.

Heuristics are usually validated against user tests, but this can be problematic as no formal method is used to validate the user test data itself. i.e., a master list of "real problems" is extracted from the user test, and the results of the heuristic evaluation are compared against this list. However, the list of real problems is extracted in an informal manner and is not reliable as a gold standard against which to compare.

Several competing sets of heuristics are available for video game evaluation, though it is unclear which are the most appropriate for use with first-person shooter games. Furthermore very little has been conducted in the way of comparative evaluation between the available heuristics. The heuristics available in the literature are the subject of a study in Chapter 4 (Testing Heuristic Evaluation for Video Games), which shows systematic problems in their use. A complete procedure to make use of the design and evaluation knowledge of heuristics is outlined in the remainder of this thesis.

2.7 Conclusion

This chapter asked the following research questions,

- What are usability and usability evaluation, and how do they relate to product lifecycle?
- What methods are used for usability evaluation in traditional domains and video games?
- How are problems discovered and analysed?
- What are the criteria and procedures for evaluating a usability evaluation method?
- What are the problems with evaluation methods documented by the literature?

Cockton (2012) considers the concept of usability, and examines some fundamental propositions about it and its development in human-computer interaction and interaction design. The chapter dismisses some naïve assumptions about an idealised notion of usability, and instead argues for a more realistic and nuanced understanding of the limitations and feasibility of the term. A number of relevant conclusions are drawn that have important implications for this thesis:

- There is no definitive answer to what usability is.
- There are no universal measures or metrics for usability.
- Methods and metrics are not completely documented in the literature.
- Usability evaluation methods do not have deterministic effects.
- Evaluator effects are to be expected.
- There are no reliable, off-the-shelf methods.
- There are no extant methods that can be immediately applied by novice evaluators.

A novel methodology is developed in Chapter 6 (The Playthrough Evaluation Framework) that takes these conclusions into account. It is based on a mixture of empirical and analytical approaches, utilising both system- and interaction-centred resources for usability problem discovery and analysis. Evaluator effects are acknowledged, but the method examines where and why they occur.

The problems explored in the literature review and summarised by Cockton (2012) are addressed by this thesis in the following specific ways:

- Development of a novel method designed for novice evaluators.
- A coherent definition of usability.
- A formal definition of the metrics and procedures involved.
- Amelioration of the evaluator effect.
- Focus on summative evaluation.

There is a relatively common nexus around which most concepts of usability are defined, but the diversity of specific definitions could be interpreted as a deficiency in the literature. Alternative, this fragmentation of the usability concept could be instructive. By breaking the construct down into constituents it should be possible to analyse usability in more detail, and with more reliability.

2.7.1 Concrete Focus on Usability

In relationship to the literature on human-computer interaction and video games, there is a critical aporia in terminology, especially for the terms user experience and player experience. This confusion exacerbates problems with the evaluator effect, which are already significant in traditional domains and evaluation methods. Game evaluation needs a solid base to build on, so this thesis will focus primarily on more concrete issues of usability. The principal contribution of this thesis is a methodology that can, in future work, be extended to address more complex issues of player experience, including individual differences in user and evaluator skill and preference.

2.7.2 Formative Evaluation Introduces More Concerns

A related issue is the additional ambiguity that is introduced in the evaluation process when considering prototype products. Formative evaluation places a great deal of emphasis on the subjective interpretation and ability of the evaluator to predict the interaction experience of a final product based on an evaluation of a prototype. While this kind of evaluation is well understood in traditional domains, no studies have explored the validity of these forms of prediction for video games. This is a concern given the complexity of the video game playing experience, their nexus of inter-related features, modality, intentional challenge, and the diversity of players. Before the validity of such formative evaluations can be tested, it is first necessary to be able to produce reliable evaluation of fixed, finished products. The field will benefit most from a strong method that can be used consistently on the same system, so this thesis will set out to make improvements to summative evaluations.

2.7.3 Improving Discovery and Analysis Reliability

Typically usability evaluation methods are compared against user test data. However, there are still many issues facing the reliability of observing and reporting user behaviour. In most cases this stage of evaluation is conducted informally, with no adequate process to test or ensure valid or reliable results. Other approaches that do provide more formal and structured procedures are designed for use with simple games for children (Baauw et al., 2005; Barendregt et al., 2003; Barr, 2010b). The overhead of using this kind of approach for a complex and fast-pace first-person shooter game would be prohibitive.

Later in this thesis procedures are developed for the discovery and analysis of usability issues in user test data. This data can then provide a reference against which usability evaluation methods can be tested.

2.7.4 Heuristics Need Clarification

Heuristic evaluation has been widely used for usability evaluation in traditional and game domains. However, this has resulted in a proliferation of competing heuristic sets, even amongst those especially designed for video games. No substantial comparison has been conducted to validate which, if any, of the available heuristics are most applicable for first-person shooter evaluation. Chapter 4 (Testing Heuristic Evaluation for Video Games) explores this question by testing 146 of the available heuristics in a large scale real world user test study.

The study also explores the problems inherent in the heuristic evaluation method itself due to its reliance on subjective interpretation. Later chapters in this thesis unpack the design and evaluation knowledge of the heuristics, and use that to derive an evaluation method that can produce more reliable results.

2.7.5 Summary

This chapter has given an overview of the literature on usability evaluation methodologies, and particularly heuristic evaluation.

The following key points are central to this thesis:

- Summative usability evaluation needs to be reliable.
- The evaluator effect is a significant problem for all domains and methods.
- More structured approaches can ameliorate it, though no current methods are appropriate for first-person shooter games.
- Heuristic evaluation offers the most domain-specificity for evaluating video games.

On the basis of this review, the following chapters develop improvements over conventional approaches. Chapter 4 (Testing Heuristic Evaluation for Video Games) shows an example of the problems evident with existing forms of heuristic evaluation when applied to the video game domain. The reasons for evaluator disagreement are further explicated in Chapter 5 (Exploring Evaluation Resource Specificity). Chapter 6 (The Playthrough Evaluation Framework) derives and presents playthrough evaluation, a novel methodology to address these problems. The method is tested in Chapter 7 (Testing Playthrough Evaluation) and shown to produce more reliable results than heuristic evaluation.

Chapter 3

Introduction to Studies

3.1 Background

During professional work conducted at Vertical Slice, several first-person shooters were evaluated, including well-known games such as *Aliens Vs. Predator* and *Crysis 2*. When our services were employed these games were almost complete, so the clients needed summative evaluations to validate their work, and to help them focus on the final few issues that remained in their development lifecycle. The evaluations of these games reported many *usability* issues, likely because the games had not been experienced by players outside the original development studios.

Developers tend to be very experienced gamers, so are likely to not experience or even able to imagine the kind of problems that novice players can encounter. This is similar to the situation with expert and novice evaluators, where novice evaluators do not have the experience observing user test sessions to be able to predict how other diverse groups of players will experience a game. This disconnection between the game authors and their audience is one of the main reasons that development studios need to employ the professional services of player researchers. The problem is further exacerbated when developers work on the same game for many months or years, and develop strategies for overcoming or working around usability problems in order to get to the content they're focussing on. As such it becomes difficult for them to notice issues that could potentially affect other novice players.

3.1.1 Formative Development Changes Rapidly

It is also pertinent to note that game content, mechanics, and controls are usually in a state of creative flux until relatively late in the development lifecycle. The interaction between many diverse multi-media elements is complex, and the final emergent consequences may not clear until they are all at a relatively high level of fidelity (McAllister and White, 2010).

During the formative stages of development developers tend to focus on creatively exploring the core player experience, and prototyping the principle mechanics that will make the game fun to play. Only once this core has been identified is usability addressed to ensure that normal players have an easy and fulfilling way to experience the underlying gameplay.

Theoretically, if production teams have a fixed plan for development that does not re-

quire substantial experimentation, prototyping, or creative development, it could be possible to schedule discrete usability tests during these formative stages. For example, if the team were certain about the controls, mechanics, level design, AI behaviour, and other interrelated design elements, it could be possible to produce minimal test cases without needing a high-fidelity advanced prototype.

However, in practice this is rarely the case. The emergent complexity of many interrelated design elements is difficult to manage, which makes it hard to predict how the final product will be. Furthermore the practical logistics of development often take precedence, with teams and projects growing or shrinking in response to changing requirements across the whole studio. This makes long-term planning especially challenging. Programming, art, and design usually take priority over user research. Investing resources to ensure good usability in the formative stages may only give temporary benefits for the particular development version being tested. The game is expected to change frequently and sometimes substantially throughout development, which may invalidate earlier efforts and resource investment.

3.1.2 Developers Focus on Usability Last

As discussed in Chapter 2 (Literature Review), usability is generally considered as a negative hygiene factor, a potential *barrier* to immersion in the core gameplay, or a quality that will *prevent* the game from being fun to play. Experienced developers have ways to bypass usability problems and immerse themselves in the core player experience that normal players do not. For example, it is common to use special "cheat codes" or "console commands" that are only available to the developers, which will launch a prototype game into key test configurations without requiring the user to perform all of the normal game interactions. Furthermore, production teams spend a great deal of time using the game system and its controls during development, and so achieve great mastery even without any explicit tutorial that a normal user would require.

As a result it is typically the case that tutorials and introductions are often not added to the game until the *Beta* stage, shortly prior to release.

Most of the games evaluated by Vertical Slice were high-fidelity, advanced interactive prototypes around this Beta stage of development. Tutorials and introductions had been created, but as they had not been tested by independent players the production teams did not have an opportunity to iterate and refine them.

3.1.3 Poetics of First-Person Shooter Games

A formal discussion of genre definitions is beyond the scope of this thesis, but there are no universally agreed definitions by the research community. Minimally a definition would include perspective from the first-person point-of-view of the player, along with the game mechanic of shooting and combat. Particularly noteworthy characteristics include an emphasis on action, speed, and player control, with often less of an emphasis on narrative concerns such as character and plot.

Functional issues tend to be critical to the player experience, due to the high volume of interaction and precision needed to quickly execute fine-detailed control.

This emphasis on the more functional poetics, particularly imitation or representation (*mimesis*) and spectacle (*opsis*), and less on the aesthetic aspects of the game also aligns more strongly with usability rather than the broader concerns of playability.

In other games with less time pressure, slower pace, and greater error-tolerance, such as RPG and strategy, player experience is less affected by these kind of usability issues, and so players are able to focus greater attention on aesthetics and diegetic qualities such as *ethos* and *mythos*.

3.2 Overview of Studies

The empirical experience of summative evaluation at Vertical Slice, and the theoretical poetics of first-person shooter games informed the motivation for the studies presented in the following chapters.

A series of evaluations were conducted using a variety of approaches: quantitative, qualitative, analytical, and empirical.

Scriven (1967) discusses different forms of evaluation, and contrasts analytical with empirical. In order to resolve this dichotomy, he suggests a mixed methods approach, called mediated evaluation. In this form, analytic evaluation precedes and informs empirical evaluation. Using a theoretically established analysis can help direct and focus an empirical study. Without this kind of framework, it can be the case that empirical studies either miss the more important areas, or their results are misinterpreted.

The novel methodology presented in this thesis, playthrough evaluation, builds on this understanding of mixed methods. In this method evaluators perform both empirical and analytical forms of evaluation. Furthermore, the method itself is derived from well established theoretical principles, giving the overall approach a well rounded and balanced basis.

Chapter 4 (Testing Heuristic Evaluation for Video Games) supports the evidence in the literature which that shows heuristic evaluation produces weak inter-rater reliability. A user test was conducted with a first-person shooter game, and a subsequent retrospective heuristic evaluation analysed the issues reported. Evaluators tended to systematically disagree on which heuristics best explained the issues. Nonetheless, principal components analysis revealed 19 core areas for evaluating these kinds of games. These components were used in latter chapters to inform the derivation of playthrough evaluation, the novel methodology presented in Chapter 6 (The Playthrough Evaluation Framework).

Chapter 5 (Exploring Evaluation Resource Specificity) presents a qualitative analysis explaining how the terminology of the heuristics and issues resulted in the weak inter-rater reliability shown in the quantitative data. Interviews revealed that evaluators were following different implicit procedures when conducting the same evaluation. Furthermore, the phrasing of specific heuristics allowed for additional deviation between evaluators' procedures and interpretations. As a consequence, a more formal procedure is needed to ensure reliable evaluation.

Chapter 6 (The Playthrough Evaluation Framework) develops a novel evaluation methodology based on the heuristics used in Chapter 4 (Testing Heuristic Evaluation for Video Games). The existing heuristics in the literature are considered to contain important and useful knowledge,

but are too ambiguous to be used reliably. A coding scheme was derived by making explicit the criteria that form each heuristic. These criteria are expressed in terms of game aspects (e.g., goals, skills, etc.), usability Outcomes (including effectiveness, efficiency, and satisfaction), and interaction Breakdowns describing the onset of usability problems.

Chapter 7 (Testing Playthrough Evaluation) empirically tests playthrough evaluation in a series of studies. Three first-person shooter games were tested with 22 evaluators. The standard metrics were computed for validity, reliability, and thoroughness. Results showed values for both problem discovery and analysis that are comparable and often greater than those seen in the literature.

Chapter 4

Testing Heuristic Evaluation for Video Games

Heuristic evaluation promises to be a low-cost usability evaluation method, but is fraught with problems of subjective interpretation, and a proliferation of competing and contradictory heuristic lists. This is particularly true for the case of first-person shooter video games, featuring complex, multi-modal, time-constrained, and intentionally challenging tasks, where no rigorous comparative validation has yet been published.

In order to validate the heuristics available in the literature, a user test of a commercial game was conducted with 6 participants in which 88 issues were identified, against which 146 heuristics were rated for relevance by 3 evaluators, summing to a total of 38,544 ratings. Weak inter-rater reliability was computed with Krippendorff's Alpha of 0.343. This weak reliability is due to the high complexity of video games and a method that does not tightly define how it is to be used. This results in evaluators interpreting different causes and solutions for the issues, and hence the wide variance in their ratings of the heuristics.

Heuristic evaluation appears to be unreliable for the summative evaluation of video game usability. Future chapters of this thesis explore why this is the case, and what can be done to reliably use the design and evaluation knowledge represented in the heuristics.

4.1 Introduction

Chapter 2 (Literature Review) outlined some of the problems found with usability and heuristic evaluation in general, which are compounded in the case of first-person shooter video games. Nonetheless, heuristic evaluation has been widely used for summative evaluation of video games. However, since the method was popularised there has been a proliferation of heuristics for evaluating this broad domain, though there is no clear way to determine which of the published heuristics are more appropriate than the others for evaluating first-person shooter's specifically.

This chapter first presents a user test in Section 4.2 (User Test) as a source of representative usability issues for this style of game. The study was part of the commercial evaluation of a video game conducted by the game usability testing studio, Vertical Slice. During the testing sessions it became apparent that the informal approach to problem detection and analysis

resulted in somewhat different reports being compiled by each of the observers. In order to explore the user test session data further a follow-up heuristic evaluation study was conducted to evaluate the use of heuristics to describe the issues recorded.

Section 4.3 (Heuristic Evaluation) shows how evaluators employed the approach from Nielsen (1994a) of rating a known set of issues from user test against a heterogenous set of heuristics. Inter-rater reliability between the evaluators proved to be systematically weak across all of the heuristics available in the literature. Reasons for this are summarised in Section 4.6 (Conclusion) which presents the argument that heuristic evaluation does not produce results with an appropriate degree of reliability for summative evaluation of video games. Section 4.4 (Validating Evaluation Themes) continues Nielsen's approach by statistically examining the evaluators' rating data using principal components analysis. Despite the low inter-evaluator reliability exhibited in the heuristic evaluation, this statistical analysis reveals underlying patterns in the evaluators' ratings which suggests that the heuristics do address to important design and evaluation themes. These themes are unpacked in subsequent chapters and used to inform the development of a novel methodology for more reliable evaluation.

4.2 User Test

Method

A single player first person shooter console game, *Aliens Vs. Predator* (Rebellion Developments, 2010), was evaluated with user testing as part of the commercial work conducted by the Vertical Slice game usability evaluation studio.

The game was a high fidelity interactive prototype evaluated shortly prior to release. Only a portion of the game was complete to a level of quality indicative of the final product, and only these sections were tested. Each session lasted approximately one hour, and the whole user test was conducted conducted over two days in laboratory conditions.

Six participants played the game on an Xbox 360 connected to widescreen HD television. Video cameras recorded the player, and realtime footage from the game console was simultaneously streamed to the observation room next door. All feeds were composed together on a widescreen HD display, and saved to disk for later analysis. The game's producer, a senior user experience consultant, and colleagues monitored the participants' play from an observation room. The user experience consultant had spent some time familiarising himself with the game before the test sessions, and the producer was able to identify when players were not playing the game as the designers had intended.

Each of the observers informally made notes on the participants' play sessions. At the end of all the user test sessions the notes were informally aggregated into a final report.

Participants

Six male players were recruited to fit the target demographic provided by the client: four self-identified "mainstream" gamers (19, 22, 20, 20 yrs) and two "core" gamers (22, 30 yrs). The mainstream gamers owned one console and played games approximately once per week, but

did not consider gaming to be a major hobby. The core gamers owned more than one console at home, played games several times per week, and self-identified as gamers.

Results

Following the user test sessions, a report was produced by three user experience professionals, including the senior consultant who ran the session. The report listed the usability and playability issues encountered by each participant, as well as some additional suggestions proposed by the senior consultant. In total 88 issues were identified (Appendix C.2.4 - Issue Analysis). While making notes and subsequently merging the individual reports it was noted that each observer had recorded issues in the test session somewhat differently to one another, occasionally attributing different aspects of the game as being problematic. These differences in interpretation suggested that it would be worth exploring the analysis in more detail by applying heuristic evaluation to the issues recorded in the final report. The main focus of this chapter is on the following stage, which was to evaluate which heuristics were violated by each issue, and hence to validate or refute their applicability beyond their original studies.

4.3 Heuristic Evaluation

In order to explore the user test data in more detail a heuristic evaluation was proposed. However, the proliferation of heuristic sets seen in the literature raises the question of which to use, and how to compare one to another. The video game heuristics in the literature were considered for suitability for testing an first-person shooter game, and those intended for different platforms (e.g., mobile,) domains (i.e., not games,) or genres (such as RTS, etc.) were excluded from further consideration, as were a number of subjective or otherwise non-validated design guidelines (Malone, 1980, 1982).

A number of lists were rejected due to being superceded (Desurvire et al., 2004), work-in-progress or not formally published in peer-reviewed venues (Desurvire and Chen, 2008; Desurvire and Wiberg, 2008; Schaffer, 2007). While Korhonen et al. (2009) created their heuristics based on evaluation of a mobile game, their structure is modular and the mobile components were removed to allow assessment of the core gameplay and game usability aspects. Similarly, the list proposed in Pinelle et al. (2008a) was based on video game reviews rather than empirical evidence derived from user testing. However, the large corpus of data from which the heuristics were extracted should serve as a solid basis for evaluation. Likewise the most recent PLAY list (Desurvire and Wiberg, 2009) was included in the study, despite being based on game reviews rather than formal user testing. The GAP list (Desurvire and Wiberg, 2010) is specifically intended to address the first experiences of game players, and was explicitly compared against user testing, so is an ideal candidate for consideration. Although not being peer-reviewed, the list in Federoff (2002) was derived from commercial game developers, so should have some practical basis, and has been significantly cited in and continues to influence subsequent academic publications. Nielsen's canonical list (Nielsen, 1994a) was included as a de-facto standard for heuristic evaluation. While it was created with data from different domains (such as productivity systems on textual and telephonic platforms) it was included in

order to compare the validation of traditional and game specific heuristics.

In some cases, heuristics from earlier publications appeared verbatim in latter sets. These exact duplicates were removed, leaving 146 unique entries (Appendix E - 146 Heuristics) remaining from the following six sources:

- Federoff (2002)
- PLAY Desurvire and Wiberg (2009)
- Pinelle et al. (2008a)
- GAP Desurvire and Wiberg (2010)
- Korhonen et al. (2009) (excluding mobile components)
- Nielsen (1994a)

Method

Three researchers who had conducted the user testing in the previous section examined each of the 88 issues reported, and considered them against the 146 heuristics.

Following the procedure used by Nielsen to derive his canonical heuristics (Nielsen, 1994a), each evaluator rated each issue against each heuristic using his 5 point ordinal scale to describe how well it explained the issue:

0. Does not explain the problem at all.
1. May superficially address some aspect of the problem.
2. Explains a small part of the problem, but there are major aspects of the problem that are not explained.
3. Explains a major part of the problem, but there are some aspects of the problem that are not explained.
4. Fairly complete explanation of why this is a usability problem, but there is still more to the problem than is explained by the heuristic.
5. Complete explanation of why this is a problem.

In total 38,544 ratings were made between the 3 evaluators, whereas Nielsen's study consisted of 25,149 ratings from a single evaluator, and where inter-rater reliability was not considered.

The three evaluators involved were a video game user experience doctoral student with professional experience of conducting user tests; a further video game user experience doctoral student (the author of this thesis), considered as a "double expert" with professional experience of conducting user experience tests and professional game development; and a further human-computer interaction researcher. The evaluators participated in a training session where the heuristics were reviewed, and uncertainty about the meaning or intention of particular heuristics was discussed and consensus reached.

The 88 issues from the user test session were randomly ordered and presented to the three independent evaluators for inspection. Every issue from the user test session was presented with the 146 heuristics, randomised in a unique way for each evaluator and each of the heuristics. This counterbalancing prevented order effects where repeated evaluation of heuristics in the same order could have influenced the evaluators' decision making process. Once each of the evaluators had completed their evaluation of all of the issues, the data were collected together for statistical analysis, presented in the following section.

4.3.1 Results

All ratings were inspected for variance between the three evaluators. Extreme variances were frequently identified in cases such as when one evaluator rated a heuristic as 5 ("Complete explanation of why this is a problem") and the other two evaluators rated as 0 ("Does not explain the problem at all"). Krippendorff's Alpha was computed across all of the ratings using an online calculator (Freelon, 2008) at a value of 0.343 (nCoders = 3; nCases = 12848; nDecisions = 38544), which represents very poor reliability. The computation was repeated with an SPSS macro from Hayes and Krippendorff (2007), and produced the same value.

It is noteworthy that in a study reported by Cockton et al. (2004), only 31% of heuristics were considered appropriately assigned. The alpha found in this present chapter is indicative of a similarly low level of appropriateness. In their later studies which employed structured reporting formats, Cockton et al. found this level increased to 60%. Similar results were reported for the percentage of problems predicted that were discovered during user testing.

4.3.2 Discussion

The systematically low levels of inter-evaluator reliability suggests fundamental inadequacies in a weakly specified, discount heuristic evaluation method when applied to the more complex scenarios in a video game user test. The original evaluation teams of the six heuristic sets considered here achieved agreement in their own studies through private discussion during evaluation. Without the decisions made during those discussions being instantiated as formal, objective evaluation processes in the methodology, repeatability and validation of their results is not possible.

A possible reason for the ambiguity in interpretation can be attributed to the different phrasing used in each heuristic set. In particular there appears to be a blurring between design guidelines and heuristics. There are a large number of design guidelines for video games, with much literature on the subject (Bateman and Boon, 2005; Fabricatore et al., 2002; Falstein and Barwood, 2006). Many of these guidelines however are tentative, subjective and informal. A reliable form of evaluation is needed to be more rigorous, measurable, actionable, and based on empirical data.

Nielsen talks about heuristic evaluation as a discount method for cases where user testing is not required, for straightforward productivity applications such as telephony and textual interfaces (Nielsen, 1992). The assumption that it is a reliable discount method needs to be considered in more detail for the case of complex video game systems.

This study has suggested that heuristic evaluation is highly subjective, subject to a substantial degree of error, and more rigorous techniques for the assessment of usability issues are needed in order to ensure this evaluation method produces valid, repeatable and valuable results. At present, heuristic evaluation was not able to be reliably validated for typical first-person shooter console video games.

The following sections explore the rating data from this study in order to understand why the evaluators assigned such different values to one another. The purpose of the investigation is to expose the implicit processes used by evaluators, and see whether there is a more reliable way to make use of the design and evaluation knowledge contained in the heuristics. Two

different approaches are used. First Section 4.4 (Validating Evaluation Themes) quantitatively examines the rating data through statistical analysis using principal components analysis, similar to Nielsen's original study. Following this, Chapter 5 (Exploring Evaluation Resource Specificity) qualitatively reflects on the evaluation process with evaluator interviews and content analysis of the heuristics in order to reveal the reasons for the evaluator disagreements.

4.4 Validating Evaluation Themes

Section 4.3 (Heuristic Evaluation) showed that individual evaluators assigned widely different ratings to the same heuristics, exhibiting low inter-rater reliability due to differences in subjective interpretation. Despite failing to be validated reliably, many of the heuristics are similar to one another and seem to address common themes.

This section reconsiders the data for further analysis, following the same approach used by Nielsen in his original study (Nielsen, 1994a). The research question explored is whether there is any cohesion in the ratings that would validate general themes for evaluation rather than specific heuristics.

4.4.1 Background

Chapter 4 (Testing Heuristic Evaluation for Video Games) was designed to be similar to Nielsen's original publication (Nielsen, 1994a) where his canonical 10 heuristics were derived. Nielsen rated many different sets of heuristics for how well they described the usability issues reported in several earlier studies. As only a single rater was used, Nielsen did not consider reliability. Instead his method was to apply a statistical approach, principal components analysis, to explore the relationships between the many different heuristics used, and to uncover underlying similarities in the themes addressed by the heuristics.

4.4.1.1 Principal Component Analysis Shows Similarities in Variables

Principal components analysis is a formally defined as a statistical method which reveals linear functions of unknown components, from a set of observed data with known variables. The analysis shows the relationships between these variables, and groups them into clusters or "components" of related areas.

The analysis shows how much of the variance in ratings is due to each of the individual components, giving an indication of the relative importance of each. In the current context, the heuristics are the variables analysed. Each variable is also described as "loading" to each component to a certain degree. This loading to a component is the correlation coefficient with the heuristic, where higher absolute values indicate stronger correlation, and negative values indicate negative correlation. Typically several similar heuristics will load strongly to single component, indicating that they all address a common theme.

4.4.1.2 Principal Component Analysis Requires Interpretation

Principal components analysis produces statistics that need to be interpreted as the method does not in itself define clear-cut conclusions, only data that need to be considered.

Heuristics that have high loadings to a particular component are more strongly correlated together, indicating that evaluators rated these heuristics similarly to one another. Low component loadings on the other hand indicate that the evaluator's ratings of the heuristics were not strongly correlated together. Every heuristic has a loading value for each component, so a decision must be made as to which heuristics are most relevant to each component. There is no universal loading threshold at which it can be said that a variable is considered to be a significant contributor to a component. Instead this decision is up to the analyst's interpretation.

Similarly the number of components to include is also a matter for interpretation, and a variety of approaches can be used.

The Kaiser criterion is the default option in SPSS and most other software. This criterion only includes components that are relatively strong candidates, indicated by a mathematical property of the component, the "eigenvalue", having a magnitude of at least 1.0 (Kaiser, 1960).

A common procedure advocated in Cattell (1966) is to visually analyse a scree plot of the components and the variance in ratings they are each responsible for. In this plot the components are ordered by decreasing amount of variance. Often there will be a number of components with higher values for variance, with a clear "step" where the plot decreases noticeably to the components that have lower variance values. This "elbow" can be used as the cut-off point at which components with lower variance values are discarded.

Alternatively, the number of components to accept can be determined by the sum of variance they collectively contribute. For example, the analyst could decide to accept the number of components that together explain 75% of variance in evaluator ratings.

A simpler approach is to decide that only a pre-determined number of components will be used, such as only accepting the top 10 components.

Regardless of any other approach used, in all cases the components should be subjectively inspected in order to determine whether they represent coherent themes. This is particularly important for components that account for smaller amounts of the total variance, and where heuristics only load weakly. In these instances it may be the case that the heuristics are not thematically related, but have been included as a component merely through the coincidental rather than meaningful co-variance in their ratings.

4.4.1.3 Nielsen's Principal Component Analysis

Nielsen's study looked at the relationships amongst 249 usability problems identified during heuristic evaluation of 11 different systems, using 101 heuristics from 7 different sets, rated by a single evaluator, summing to 25,149 individual evaluations. The study found that 53 components were needed to explain 90% of the variance amongst ratings of the heuristics. 25 components accounted for 1% or more of the variance, summing to 62% of the variance in total. There was a gradual decline of variances in the scree plot, so no clear cut-off point was identified. Instead the top 7 components that accounted for the most variance (30% of the total) were arbitrarily chosen to be included in the analysis. The loaded variables were examined, and components were given descriptive names to represent the common areas they addressed. The top 7 components and the amount of variance they accounted for is shown in Table 4.1 ("Nielsen's 7 components and variances") on the next page

Table 4.1: Nielsen's 7 components and variances

Variance	Component
6.1%	Visibility of System Status.
5.9%	Match between system and real world.
4.6%	User control and freedom.
4.2%	Consistency and standards.
3.7%	Error prevention.
3.1%	Recognition rather than recall.
2.8%	Flexibility and Efficiency of use.

These components were then used to inform the derivation of his canonical set of heuristics. A similar approach was used in the study presented in this chapter. Principal components analysis was used to identify the underlying themes addressed by the heuristics rated in Section 4.3 (Heuristic Evaluation). These themes were then used to derive novel evaluation resources in later chapters of this thesis.

4.4.2 Method

Principal components analysis was conducted on the heuristic rating data from Section 4.3 (Heuristic Evaluation). In that study 88 usability issues were considered within one system, employing 149 heuristics from 6 different sets, and rated by 3 evaluators, summing to 38,544 ratings in total. Following the recommendation from Kaiser (1960), Varimax rotation was applied, as was Kaiser Normalisation. Additionally, as suggested by Stevens (2002), components were only considered if they loaded with absolute magnitudes of 0.40 or greater. The software used was SPSS Statistics 19.

Three separate analyses were produced, one for each of the three evaluators, as well as a composite of all of the evaluators' aggregated data.

4.4.3 Results

The principal components analysis shows how the evaluators' ratings of each issue vary with respect to the other heuristics, revealing underlying themes that are common across separate sets of heuristics.

Setting Thresholds

In order to identify how many components should be included in the analysis, scree plots were produced. Appendix A.3 (Scree Plots) presents the plots for each individual evaluator and the aggregate of all three evaluators' ratings. As with Nielsen's data, in each case the graphs show similar results with a gradual decline in the slope of the curve. There is no characteristic "elbow" or drop-off in the variance contributed by each component that could indicate a threshold for including components in the analysis.

As no clear threshold was seen in the plots it was necessary to reflect on the heuristics included in each component in order to determine which components represented a coherent theme. Generally the components that contributed greater amounts of the total variance in ratings included heuristics that were similar to one another. Components with lower variances tended to include heuristics that were unrelated. The analysis produced these components as the ratings of these heuristics were coincidentally similar, but not in a meaningful way.

Identifying Similar Components

Each component produced by the analysis consists of multiple heuristics that tended towards being given the same rating by the evaluator. Although the three evaluators ratings exhibited low inter-rater reliability in the heuristic evaluation, the components identified by the principal components analysis show similar groupings of related heuristics.

In order to make sense of the data the variables loading on each component need to be inspected and a descriptive name assigned to represent the area they address. For example, the component that generally accounted for the greatest amount of variance brings together heuristics related to skills that the player learns during the game. As such this component was given the descriptive name, "Learning Skills".

There were noticeable similarities in the components identified for each evaluator, so in the next stage of analysis similar components were grouped together. For example, all three evaluators had components that addressed the player's ability to learn and use skills needed to accomplish game tasks. To demonstrate these similarities, loadings for the heuristics associated to these components are shown in Appendix A.5 (Component Loadings). These tables show the heuristic loadings for the component, "Learning Skills". Loadings are listed for each of the individual evaluators' components that addressed this theme as well as a separate figure for the heuristic loadings on the component aggregated from all three evaluators' ratings. All three evaluators had components that deal with the issue of learning skills, and in most cases the same heuristics appeared in each component, and often in the same order. For example, Heuristic 68: "Player given opportunity to model correct behavior and skills" contributed the second most variance of the ratings for evaluator 1, and the most variance for evaluator 3 and evaluator 2.

Drawing out Meaning in the Components

The heuristics loading on these grouped components were inspected in order to determine which brought together heuristics that were coherently related to one another, and which may have be included through random chance. 21 meaningful components were identified across the three analyses. The heuristics further influencing the remainder of the components were increasingly unrelated to one another, and the amount of variance they contributed continued to decrease. Descriptive names were assigned to represent the areas addressed by the related heuristics. Appendix A.4 (Component Variance) shows the descriptive names for these components, along with the amount of variance in ratings that they each explain. Separate tables are presented for the individual evaluators' ratings, as well as the aggregate of all three evaluators' ratings.

The aggregate analysis found that 37 components explain 77.10 % of the total variance in the ratings. The final 21 components identified collectively account for 57.85% of the variance, each of which contributes greater than 1% to the total. The top 5 components each explaining more than 3% of the variance account for 22.49% of the total variance. These figures are in a similar range to those reported by Nielsen.

4.5 Discussion

As can be seen by comparing Table 4.1 ("Nielsen's 7 components and variances") on page 90 and the tables in Appendix A.4 (Component Variance) there is only a small amount of overlap in the component analysis between Nielsen's original list and this present study. This is clearly due to the differences in domain, requirements upon the user, and purpose of the system. It is interesting to note that the percentage of variance explained by the top seven components are similar in each list.

4.5.1 Principal Component Analysis Suggests Core Themes to Evaluate

The components identified in this study represent the principal themes for evaluating first-person shooter games and accord with other informal definitions of first-person shooter seen elsewhere in the literature (Fabricatore, 1999; Pinelle et al., 2008b).

As mentioned earlier the kind of *functional* qualities associated with this kind of game include a high degree of action, speed, and player control, with less concern for *diegetic* qualities emphasised in most player experience studies, such as narrative, character, and plot. This can be seen reflected in the fact that of the 21 components identified, the least significant 2 address these aspects that are more common in traditional, non-interactive media. The heuristics associated with these two components only received the lowest ratings in the heuristic evaluation in this chapter, and in general were not considered to be strongly applicable to first-person shooter games in general. They were discarded from further analysis, leaving 19 components that address fundamental usability issues for this genre.

Principal components analysis reveals that in broad terms, the 146 heuristics reviewed from the literature attempt to address the same core areas. This is encouraging, though the field is not helped by a proliferation of variously phrased versions. Despite these shortcomings and differences between evaluators' interpretations, there is consistency in the resultant components. Furthermore, the analysis also suggests a number of areas for evaluation. Rather than adopting the received wisdom that around 10 heuristics are appropriate, this study demonstrated that a core of around 19 components are relevant for first-person shooter games.

The evaluators in Korhonen et al. (2009) commented that they found one of the lists (called HEP) to have too many heuristics (43) though no suggestion is given for what an appropriate number would be. The principal components analysis in the present study suggests that there are around 20 distinct components which are candidates for derivation as heuristics. Furthermore, the evaluators in Korhonen et al. (2009) reported that heuristics from both HEP and their own set suffer from problems due to an inappropriate abstraction level - that is they are either too specific (likely in the case of HEP) or too general (more likely for Korhonen et al.) In the

latter case the issue is that they are of limited use in guiding and assisting the evaluation. It is interesting to note that one of their evaluation teams left 16% of issues unassigned to any of their own heuristics, and the other team left 30% unassigned to any from HEP. In the present study, there was at least one heuristic from Korhonen et al. that was rated as 3 or above for only 16 of the 88 issues (18.18%), meaning that 81.82% of issues were not addressed by any of heuristics from that set (i.e., they would have been reported as unassigned following their method.) As such, 10 heuristics may be too few as they lack sufficient specificity to explain the issues encountered.

Nielsen's original 10 heuristics were included in this study as a benchmark for usability only, but they also exhibited a similar degree of ambiguity and lack of reliability as the others. While they may be useful for identifying general areas for heuristic evaluation it has been shown they are equally vulnerable to the evaluator effect as the others. What's more it is noteworthy that Nielsen (1994a) did not take into account the inherent subjectivity of interpretation involved in matching heuristics to issues as they did not involve any other evaluators hence there was no inter-rater reliability measure.

4.6 Conclusion

From a user test session of a commercial first person shooter console video game, 88 issues were identified and rated for degree of explanation against 146 heuristics by 3 evaluators, totalling 38,544 individual ratings. Inter-rater reliability was computed using Krippendorff's Alpha at a low level of 0.3429, suggesting systematic problems with the method.

Heuristic evaluation has promise as a low-cost usability evaluation method, but is fraught with problems of evaluator disagreement, and a proliferation of competing and contradictory heuristic lists. This is particularly true in the field of games research where no rigorous comparative validation has yet been published.

The weak reliability seen in this study appears to be related to the high complexity of video games and a lack of structure and clear criteria to guide the evaluation, resulting in evaluators interpreting different reasonable causes and solutions for the issues, and hence the wide variance in their ratings of the heuristics.

Subsequently principal components analysis was used following Nielsen's approach in which he derived his canonical 10 heuristics. The analysis revealed that despite disagreeing over specific heuristic ratings, 19 underlying components were extracted, validating the general areas addressed by the separate heuristics. This suggests some coherence in the general heuristic areas, even while the specific heuristics themselves exhibited weak inter-rater reliability. This is an important novel contribution as - at least within the current scope of first-person shooter games - it provides evidence contrary to the received wisdom that 10 heuristic categories are sufficient. The literature has produced a proliferation of new heuristics, but this study shows that they can be seen as cases of triangulation around the same general areas. Nielsen's use of this analysis was to derive his canonical set of heuristics, but this study suggests that more heuristics are not needed as they are likely to continue to circle around the same core areas but still exhibit the same weak reliability. Instead, what is needed is a more nuanced and reliable

means to evaluate these core areas. These components represent the areas that an evaluation needs to consider, but they still need to be operationalised into a form suitable for use in an evaluation. The analysis conducted in this chapter validated a core set of 19 components as being candidates for development as a novel evaluation resource in a future study. The term "candidate" here is important. In Nielsen's original study, though he had derived 10 components and presented them as heuristics, he was also clear to point out that it remained to be seen whether they were appropriate for use as heuristics.

Subsequent chapters in this thesis make use of the data presented here as the underlying components relevant to first-person shooter usability evaluation. While the identification of the core components is an important step towards evaluation, it alone does not address how they should be used. Chapter 5 (Exploring Evaluation Resource Specificity) considers the components in detail, and unpacks the issues and heuristics associated in order to restructure and improve the design and evaluation knowledge originally represented in the ambiguous heuristics. In Chapter 6 (The Playthrough Evaluation Framework) this unpacking is then used to inform a method for conducting evaluation in a more testable and reliable way. A novel evaluation method, playthrough evaluation, is then presented, which shows how these restructured resources can be used. Chapter 7 (Testing Playthrough Evaluation) applies the playthrough evaluation method, compares it to heuristic evaluation, and demonstrates that the restructured resources and methodology provide improvements to inter-evaluator reliability.

Chapter 5

Exploring Evaluation Resource Specificity

5.1 Introduction

Chapter 4 (Testing Heuristic Evaluation for Video Games) revealed poor heuristic inter-rater reliability, but this was further explored to reveal consistent underlying themes addressed by the heuristics. Given that underlying consistency, this chapter considers the content and presentation of heuristics in order to understand whether they could be repurposed in a more reliable form.

Section 5.3 (Unpacking Evaluator Interpretations of Complex Issues) provides a detailed example which demonstrates three different emphases that heuristics can focus on. Section 5.4 (Heuristic Types) presents insights into why the low inter-rater agreement occurred in Chapter 4 (Testing Heuristic Evaluation for Video Games), showing how the specific phrasing of heuristics contributes to the variability in evaluators' decision making process.

5.2 Background

5.2.1 The Evaluator Effect Throughout Usability Evaluation

As discussed in Chapter 2 (Literature Review), problems with reliability are introduced at many stages during the evaluation process, from issue identification through to analysis and categorisation.

Errors in evaluation that impair reliability are initially introduced at the observation stage, where different evaluators do not even agree strongly on the observed events. This is especially true for retrospective freeform heuristic evaluation, where evaluators do not follow a formal procedure to detect problems. In this form, heuristics are only referred to once the evaluator has already decided that some kind of problem has occurred. In these cases where the problem incidents are chosen in a freeform manner, evaluators typically don't identify the same issues in a game as being problematic, let alone rate or categorise them the same. In an evaluation of a simple and relatively static children's game, Barendregt (2006) showed that for a fixed list

of *pre-defined* observation points, however, reliability does increase over the more realistic but demanding case where evaluators had to first detect the observation points themselves.

5.2.2 First-Person Shooter Action Games Involve Complex Issues

Even with heuristic evaluation adaptations that employ more structured approaches, the types of typical user task are quite different to the context in which users experience tasks in a video game. For example, Schmettow and Niebuhr (2007) used three representative tasks that are common in a bibliographic tool,

- add reference.
- search for reference.
- export list of references.

Typical engagements in fast-paced first-person shooter games are considerably more complex, multi-modal affairs. In the studies presented in this thesis players were sometimes not even aware that the game had given them a new task. This can clearly be as a result of poor usability in respect to instructing the player, though often it is an entirely intentional practice, leaving the player with some mystery as to what is actually needed, and requiring series of actions that could involve navigating and exploring 3D environments, interacting with combinations of objects, solving logic or physics puzzles, negotiating with AI, etc.

Cockton and Lavery (1999) point out that for any given issue, it is unlikely that a single cause can be justified without extensive experimentation to prove that it is the reason for the issue. This appears especially true for video games, where the complexity of dynamic interaction, with often a fast-paced and multi-media interface, rapidly increase the difficulty in identifying a single prime cause. There are almost always multiple possible causes and hence solutions for any given issue.

Complex Issues Are Difficult to Evaluate Reliably

While the evaluator effect has been widely discussed, there is little detailed understanding of specifically how and why it occurs. Cockton and Woolrych (2001) propose a taxonomy of task complexity that goes some way to help understanding these disagreements. In their terminology the simplest usability issues are immediately "perceivable" just by observing the system. The next level of complexity is "actionable", which requires a few small interaction steps such as button clicks to reveal the problem. The final level is "constructable", which are usability problems that require several steps to be discovered. Most of the issues that had high levels of disagreement involved many complex steps, involving several different aspects of the system. Section 5.3 (Unpacking Evaluator Interpretations of Complex Issues) explores how and why the evaluators rated the heuristics in the way they did.

For example, during a gun fight the player was reminded how to use a combat skill that was taught earlier, but in the heat of the action they responded by pressing the wrong button. In this case it was reasonable for different evaluators to identify this as being a problem with the game controls, with the tutorial, or with the reminder presented to the player. As such their ratings for each of the heuristics corresponding to each of these areas were different.

The evaluators showed clear disagreement in numerous other complex scenarios that involved many interrelated factors.

It's worth noting that in a heuristic evaluation presented in Cockton et al. (2004) only 31% of heuristics were correctly assigned by novice evaluators. Krippendorff's Alpha of 0.343 computed in Chapter 4 (Testing Heuristic Evaluation for Video Games) is indicative of a similarly low level. In the later studies of Cockton et al. (2004) which employed structured reporting formats, this level increased to 60%. Similar results were reported for the percentage of problems predicted that were discovered during user testing. In another paper (Cockton and Woolrych, 2001) show that in non-trivial tasks most heuristics were incorrectly used, and that these errors increase with task complexity.

The weak inter-rater reliability in Chapter 4 (Testing Heuristic Evaluation for Video Games) appears to be related to the complex, "constructable" nature of the tasks that simultaneously involve several different aspects of the system. During the evaluation each evaluator focussed on different aspects of the issues and heuristics, and so assigned different ratings to one another. The steps needed to discover the problem were not sufficiently small, atomic actions to belong to the simpler *perceivable* or *actionable* categories, which could have been evaluated with greater specificity and reliability.

5.2.2.1 Explicit Structure Facilitates Inspectability

While informal methods like heuristic evaluation suffer from poor reliability, more structured and formal methods demonstrate improvements in inter-evaluator reliability as described in detail in Section 2.4 (Evaluator Effect). By explicitly separating the stages of evaluation and defining clearer discovery and analysis resources, evaluators identify issues more reliably. The explicit and exposed nature of the process furthermore facilities introspection and inspectability.

In traditional usability evaluation methods like heuristic evaluation, groups of evaluators tend use private group discussions to resolve any disagreements. Unfortunately, though, the design and evaluation knowledge raised during these meetings is not reported in the literature, and so not available to benefit other research groups or methodologies. Playthrough evaluation differs in this regard by providing a highly structured and explicit process that can be critiqued in detail, and made available for the research community to discuss and improve on. Heuristic evaluation studies, in contrast, generally only publish the end result, which is usually just another distilled list of novel heuristics. These heuristics attempt to condense design and evaluation knowledge into a compact form, which is very useful for convenient light-weight use. However, in the process of condensing and distilling, there is necessarily a loss of information.

In order to address these and other related troubles, Lavery et al. (1997) propose that reports of usability issues describe separate aspects of a problem:

- Cause
- Breakdown
- Outcome

After defining their structured problem report, they propose that it will then be possible to define rules for matching different issues together, although this is something that they leave for future work. They identify three questions that need to be answered:

- Which components of a usability problem report are used for the matching?
- How should blank components of a problem report be addressed?
- How are disagreements in one component addressed, when other components match?
- When do two components match?

To answer these questions they propose that multiple measures of agreement be used, such as the number of components that match, rather than a single binary match / no match decision. These questions are returned to in Chapter 6 (The Playthrough Evaluation Framework) where playthrough evaluation is described, including ways to quantify the degree of matching between independent problem reports. Further discussion is presented in Chapter 7 (Testing Playthrough Evaluation), an empirical study testing the method and reporting on the inter-evaluator reliability it produced.

The separation of problems into various stages, including breakdown and outcome, is taken on in playthrough evaluation with some subtle changes. In Lavery et al. (1997), evaluators were required to judge the cause of the breakdown, but had no analysis resources available to help make this assessment. This is similar to heuristic evaluation where evaluators identify different heuristics as describing a problem because they identify different causes. While the separation of cause from breakdown and outcome is useful in that it helps to distinguish where the evaluators disagree, it is actually introducing new opportunities for disagreement by encouraging more subjective and informal judgements.

5.3 Unpacking Evaluator Interpretations of Complex Issues

In order to explore the disagreements further, the evaluators from Chapter 4 (Testing Heuristic Evaluation for Video Games) individually participated in semi-structured interviews to discuss the decision making process for their ratings. A selection of representative ratings that exhibited particularly high levels of disagreement were chosen for the subject of the interview. The evaluators were asked to comment on how and why they assigned their ratings for these cases. Evaluators tended to see the causes of issues from a slightly different perspective to one another, but it was clear that there were multiple reasonable interpretations for which heuristic best explained each issue. This appears to be due in part to the complexity of the tasks involved and the ambiguity of the heuristics used.

5.3.1 Poor Heuristic Specificity Diminishes Reliability

In Chapter 4 (Testing Heuristic Evaluation for Video Games) several particularly interesting cases of weak inter-rater reliability occurred that were related to the poor specificity of the heuristics. Although the evaluators chose a particular heuristic as the best explanation for an issue, this

was only the best choice from the limited set of heuristics given. It may not be a very good objective explanation for the issue, though it was the best available.

For example, Issue 16: "There are three separate textual messages on the screen simultaneously" relates to the breakdown in the player's ability to notice the messages on screen. None of the 146 heuristics from Chapter 4 (Testing Heuristic Evaluation for Video Games) were rated highly as explaining this issue. The closest match was Heuristic 92: "Screen layout is efficient, integrated, and visually pleasing" yet it only received a mean score of 2.33 out of a possible 5.0 across all 3 evaluators.

Evaluator 3 rated this heuristic higher than any others at level 4, as a broad interpretation of the heuristic would accept that it deals with the presentation of information on screen in general. Evaluator 2 however assigned a rating of level 0 to the same heuristic as the problem was not to do with screen layout *per se* and was more a question of Effectiveness than Efficiency, and evaluator 1 gave a moderate rating of 3. Evaluator 1 gave a different heuristic the highest rating of 5 for this issue, Heuristic 36: "Game provides feedback and reacts in a consistent, immediate, challenging and exciting way to the players' actions". However both evaluator 3 and evaluator 2 rated that heuristic at level 0 for this issue. The text displays are indeed a form of feedback, though these two evaluators still did not feel that any of the specific criteria in the heuristic had been violated.

Evaluator 2 rated all but two heuristics at level 0, assigning the highest rating of 4 to Heuristic 100: "The game does not put an unnecessary burden on the player" whereas evaluator 1 rated this heuristic at level 0 and evaluator 3 rated it at level 2. The player was burdened in the sense of dealing with information "overload", but the other evaluators did not recognise this heuristic as being specific enough to explain this issue.

The ratings for this issue show three similar but different interpretations, one for each of the evaluators:

1. Feedback presentation.
2. Visual processing burden
3. On-screen information.

A reliable usability evaluation method should ideally facilitate the same correct interpretation from all evaluators. This is the purpose of the player action framework.

In many cases the evaluators disagreed over the specific rating to apply to each heuristic for each issue. Often one evaluator took a more liberal interpretation of the heuristic text when considering whether it described the issue.

For example, the evaluators considered Issue 13: "Player comments almost immediately that 'There's this thing at the bottom showing me which way to go.'", in which the player refers to the waypoint indicator that shows the player which direction they need to travel in to reach their next objective. Evaluator 2 and evaluator 3 both assigned a rating of level 0 to Heuristic 94: "Status score Indicators are seamless, obvious, available and do not interfere with game play". The heuristics does not explain what "status score indicators" really means, and the evaluators did not agree that the waypoint indicator was a "status score" indicator. Evaluator 1, however, felt that a less literal interpretation would be useful, and rated the heuristic at level 5. In this

case the decision hangs purely on a semantic issue of the whether this user interface element is of the type described. The waypoint indicator does not match the description of a "score ...indicator" exactly, and whether it indicates "status" or not remains a moot point.

Status was taken by evaluator 1 and evaluator 2 to mean things like the currently selected weapon, remaining health, etc. However, taking too much of a literal interpretation of the text may not be the most useful approach when conducting an evaluation. If we accept a less rigid definition of "status" as including the current task status, then we begin to move towards an understanding that would be relevant for this issue. What's more, by expanding the scope of this heuristic to include such elements, it potentially increases the discovery and analysis resources available during the evaluation. Specifically this means that the waypoint indicator is now included in a systematic evaluation that enquires about whether it is "seamless", "obvious", "available", and "interferes with play". None of these analysis resource terms are specified in other heuristics. If the waypoint indicator had not been considered by this heuristic, then these terms would not have been included in the evaluation procedure. As such, potential issues may be missed or incorrectly evaluated.

This example has shown that it is useful to consider how the discovery and analysis resources could apply, even in cases where a terse heuristic summary does not explicitly include them.

5.3.2 Rating Analysis

This section considers a representative example issue, and presents a detailed examination of the evaluator ratings as exemplifying three different approaches to heuristic design.

The following observation was recorded during post-session interview,

> Issue 39: "The player says he didn't know that he had the plasmacaster or know how to select it. He also didn't realize there was an energy or health meter"

The *Plasmacaster* is a weapon that the player had picked up earlier in the game, and the meters are visible indicators of his character's health and energy. Table 5.1 ("Heuristic Ratings for Plasmacaster") on the next page lists all of the heuristics for this issue which were rated at level 4 or greater ("Fairly complete explanation of why this is a usability problem, but there is still more to the problem than is explained by the heuristic.") by at least one of the three evaluators, preceded by ratings for each evaluator: Evaluator 1, Evaluator 2, Evaluator 3,

5.3.3 Three Different Interpretations

The three evaluators' comments regarding their ratings can be summarised as follows: Evaluator 1 felt that the tutorial had not trained the player in the skills necessary to understand that he had collected the Plasmacaster, or what the health and energy meters were. While this may be the case, it doesn't help to identify specifically what was wrong, and only goes so far as to imply that an idealised tutorial would have helped the player to develop the necessary skills. Clearly these heuristics are *design principles* that no one would disagree with, but as such do not provide much value in terms of identifying the cause of the problem.

Table 5.1: Heuristic Ratings for Plasmacaster

Evaluator 1	Evaluator 2	Evaluator 3	Heuristic
5	0	2	Heuristic 136: "The skills needed to attain goals are taught early enough to play or use later, or right before the new skill is needed"
5	0	2	Heuristic 95: "Teach skills early that you expect the players to use later"
4	0	0	Heuristic 57: "Player able to demonstrate and practice new actions without severe consequences. Player knows what actions to take"
4	0	2	Heuristic 68: "Player given opportunity to model correct behavior and skills"
4	0	2	Heuristic 74: "Player provided with opportunities to practice new skills so as to commit skills to memory"
2	4	0	Heuristic 99: "The game contains help"
3	4	1	Heuristic 83: "Provide instructions, training, and help"

Evaluator 2, however, felt that in a formal sense none of the explicit criteria dealing with training had been violated as the game had in fact presented the player with a tutorial. These ratings suggest an alternative solution: if the player were given the ability to refer back to the instructions through a help system, they would be able to refresh their training whenever needed. This may be particularly relevant if the player takes a long break from the game, and cannot remember the training when they return. While this may be a reasonable idea which avoids the overhead of redesigning the tutorial, it still lacks clarity and does not contribute to an understanding of what had caused the problem.

The ratings of Evaluator 3 were low for all of the heuristics as none of them explained the underlying *cause* of the problem. Additionally, Evaluator 2 and Evaluator 3 both felt that the term "skills" only meant components that the designers had intended to require skill, such as tactical mastery of game mechanics or manual dexterity with controls, as opposed to the general capability of visually parsing the screen, and of cognitively understanding the meaning of the HUD. Furthermore as the player successfully completed the level without any apparent negative *outcomes*, it was not clear whether the issue should legitimately be considered to be a *problem* for this user session.

Common Themes Despite Disagreement

Considered together, these heuristics tend to deal with the player's ability to understand and execute the right actions, though we see considerable disagreements in ratings when it comes to the specific phrasing involved. An apparently pragmatic conclusion would be that these ratings are sufficiently similar to justify merging them into a composite heuristic, which traditionally is what would occur in a private group discussion between evaluators. As a broad summative evaluation (as a game review, for example), this high level of abstraction might perhaps be sufficient. However, the evaluation conclusion becomes less useful as a way to identify and understand the causes and outcomes that affect real players.

Heuristic Mix Cause, Effect, and Guidelines

The evaluators' disagreements suggest a number of ambiguities with the current heuristic evaluation method. For example, these evaluations raise questions of how to decide whether the game has provided the player with sufficient skills tuition, and indeed, what the skills are, and to what game elements they apply. These questions are intentionally not answered by the heuristics or the methodology itself, as it was originally intended to be used as a high-level expert review of simple prototype systems. However, this ambiguity is also the source of low reliability, especially when dealing with the more rigorous requirements of summative evaluation, and the complexity of first-person shooter games.

As introduced earlier in Chapter 2 (Literature Review), Hollnagel (1993a,b) make an important distinction between *"phenotype"* and *"genotype"*, i.e., between the underlying cause of a problem and the observable outcomes. He described several different taxonomies of error but critiqued them where they mixed empirically observable phenomena with subjectively inferred influences. This is a key concern for this thesis, and a similar critique can be levelled broadly at the heuristics in the literature. Later stages of this thesis, particularly in Chapter 6 (The Playthrough Evaluation Framework), operationalise the design and evaluation knowledge contained in these heuristics, and do so in such a way as to cleanly separate breakdowns and outcomes. The consequence is to improve inter-evaluator reliability of observable outcomes, and therefore to allow for better understanding of the evaluator effect with regards to the quality of inferences.

This chapter considers how to define explicit criteria for each of the heuristics, and Chapter 6 (The Playthrough Evaluation Framework) develops this idea further into a coding scheme that can be used to evaluate a play session. By comparing this standard terminology and format for heuristic criteria and issue transcription, it becomes straightforward for evaluators to all agree on whether the criteria have been conformed to or violated for each issue.

5.3.4 Separating Cause and Effect from Composite Heuristics

Several of the heuristics considered exhibited aspects of all three forms mentioned earlier, merging design principles with outcomes and breakdowns. As a result, evaluators disagreed about what would constitute violation or conformance to the heuristic. Note that heuristic evaluation has been criticised in the literature (Cockton, Woolrych, Hall, et al., 2003) for finding a large number of false positives, which may be due to the separation of breakdowns from outcomes. An evaluator could reasonably decide that a heuristic describes a breakdown that could occur, without giving appropriate consideration to outcome or vice-versa. In order to ameliorate this effect, Section 5.5 (Heuristic Unpacking) develops a method which explicitly identifies criteria for breakdown and outcome. Furthermore, once each of these individual aspects are evaluated separately, they are then combined in order to consider whether a possible breakdown is likely to result in a significant outcome. That is to say that any potential breakdown has to be described with likely outcomes too. It is not sufficient to note that a potential breakdown could occur, and use that as justification for heuristic violation. In the new method, breakdowns are considered initially for whether they are likely to occur, and what the consequential outcome may be.

5.4 Heuristic Types

These three different emphases (causes, outcomes and design principles) point to an important observation: that the term "heuristic" is used in several different, and potentially contradictory ways. This is supported by the observations from Doubleday et al. (1997) who notice a lack of "evenness" in heuristics. When considering Nielsen's canonical set, they comment that some are simple and precise, yet others are "imprecise and difficult to check for completeness". This thesis goes further by identifying three distinct types of heuristics seen in the 146 used in Chapter 4 (Testing Heuristic Evaluation for Video Games), and defining novel terminology to distinguish their use:

- Design principles.
- Abstract reflection.
- Outcome analysis.

5.4.1 Heuristics as Design Principles

The derivation of heuristics as *design principles* may be useful for distinguishing between games ranked high or low by professional or consumer opinion (Desurvire and Wiberg, 2009; Pinelle et al., 2008a). As such they may also have applicability as formative guidelines to assist designers during pre-production. For example, Heuristic 81: "Provide consistent responses to the users actions". However, these heuristics were not validated so have limited applicability for use as evaluation tools. Furthermore, Grudin (1989) points out that abstract design principles such as "consistency" lack actionable definitions with which to guide development and to differentiate between good and bad cases. Korhonen et al. (2009) likewise reflect on heuristic specificity with similar concerns regarding inappropriate abstraction levels.

Polson et al. (1992) argue against the use of design guidelines such as "minimize working memory load" by pointing out that no means to measure working memory is specified, nor are solutions proposed which could ameliorate the problem. Similarly they consider heuristics such as "Use simple and natural dialog" less useful than a cognitive walkthrough analysis which can guide evaluators in understanding why a problem has occurred and how to remedy it. They conclude that such simplified guidelines have little to contribute to complex interactions.

5.4.2 Heuristics for Abstract Reflection

Nielsen's use of the term "heuristic" does has some applicability in formative and summative evaluation contexts. However, these types of heuristic feature the most abstract phrasing, referring to general areas for the evaluator to consider but without defining specific criteria for violation of conformance. For example, Heuristic 146: "Visibility of System Status". These *abstract reflective* forms mirror the way in which they were created through principal components analysis, a dimension reduction technique which was used to reveal implicit similarities among 101 different heuristics, and to reduce them to a more abstract set of 10. This high level of abstraction means that they still suffer from ambiguous specificity and weak inter-rater reliability when used as evaluation tools.

5.4.3 Heuristics for Outcome Analysis

Other heuristics validated against user testing may be more specific, with clear criteria for violation or conformance, and hence exhibit greater inter-rater reliability. For example, Heuristic 142: "There should be variable difficulty level". This form of phrasing is particularly suitable for *analytical outcome* based evaluation, especially in respect of the standard usability aspects of Effectiveness, Efficiency and Satisfaction. However, they still do not address how these three criteria are affected by particular design decisions, so contribute little to design knowledge about causes of problems.

5.4.4 Analytical Heuristics Provide More Specific Evaluator Resources

Many of the heuristics considered take the analytic form, which make them readily available for decomposition into breakdowns and outcomes. Those heuristics which take more of a design principle form, or even more so for the abstract/reflective type, are considerably more general and ambiguous. They map less clearly to specific, observable breakdowns and outcomes.

Consider, for example, Heuristic 78: "Players should be given context sensitive help while playing so that they are not stuck and need to rely on a manual for help". It would be difficult to argue against this principle in the general form: "players should NOT be given context sensitive help while playing so that they ARE stuck and need to rely on a manual for help"[1]. So the question remains, how to operationalise this heuristic? It would be excessive to expect all contexts to provide unique help. Alternatively, it is unlikely that a game could dynamically detect when the player is genuinely stuck, and then provide context sensitive help for that specific issue. Furthermore, there are myriad reasons why a player could become stuck. This heuristic does not deal with detecting and resolving these underlying problems *per se*, but rather proposes a workaround, assuming that the problem may occur without considering why.

In terms of analysis and discovery resources then, this heuristic only addresses discovery in terms of observable consequences of an actual user test. It has little to contribute in terms of discovery for a prediction of when a problem could occur. As it does not describe any underlying causes which would trigger the observable outcomes it can only be used to discover the outcomes of a problem that has already occurred. Thus, this heuristic also provides little in the way of analysis resources. As a result this means that it becomes difficult to explicitly define the analytical Breakdown criteria for the heuristic's violation. However, it is clear what the Outcome criteria will be, particularly for Efficiency and Satisfaction, and also what other kind of observable events can be used to indicate violation (e.g., the player tries to find help in the game manual).

In several cases, even heuristics which take the more empirically measurable **analytic** form still exhibited low reliability in the heuristic evaluation. For example, Heuristic 36: "Game provides feedback and reacts in a consistent, immediate, challenging and exciting way to the players' actions". This is a good example of a heuristic with many disparate component criteria. This introduces the potential for disagreements amongst evaluators when deciding whether the

[1] Clearly challenge is an important part of first-person shooter games, but being "stuck" is more likely to be experienced by the player as excessive or inappropriate challenge, with negative outcomes for Efficiency and Satisfaction.

heuristic as a whole has been violated based on a subset of criteria violations. For example, one evaluator may feel that the heuristic has been violated due to the game being too easy (a violation of the criterion "Game ... reacts in a ... challenging ... way to the players' actions"). Another evaluator may feel that the level of challenge was appropriate, but that the heuristic is violated due to some *inconsistency* in the game, entirely separate to the issue of challenge. In these cases the heuristic may be rated as often being violated, but with a rather low rating. Nielsen's heuristics were a good example of these kinds of ratings, high frequency but low specificity.

Restructuring Design and Evaluation Knowledge

Rather than derive new heuristics, this current chapter instead focuses on operationalising the existing ones by requiring evaluators to rate the criteria which constitute the heuristics, rather than the composite heuristic itself. In particular Outcomes and Breakdowns are separated to assist the evaluator in understanding what has occurred, how the problem may have been caused, and the resulting effect it produced. This approach has a similar aim to that proposed by Matera et al. (2002), but the particular techniques involved are more transparent. In Matera's approach, "Abstract Tasks" were informally derived from experts' opinion about how to conduct an evaluation. They consist of concrete steps that a novice evaluator can conduct, and a complete set of Abstract Tasks defines a systematic procedure for evaluating a complete system. The framework presented in this current thesis makes use of existing heuristics from the literature as the source of points for the evaluator to consider, similar to Matera's way of using expert opinion as the source from which Abstract Tasks are derived. From these existing heuristics are derived explicit, concrete means to measure conformance or violation

5.5 Heuristic Unpacking

We have seen how the term heuristic is used in a variety of ways, phrased with different degrees of abstraction, and how each of these different forms is most suitable for different purposes. Furthermore, it has been shown that this degree of abstraction affects the extent of inter-rater reliability.

The separation of conventional usability outcomes and interaction breakdowns suggests a means to thoroughly examine the criteria explicitly and implicitly defined in each heuristic. In the following sections an approach to deconstruct an example heuristic is presented, and it is shown how this can help explain and improve the poor inter-rater reliability in Chapter 4 (Testing Heuristic Evaluation for Video Games).

The following section introduces content analysis as a method for unpacking the design and evaluation knowledge contained within heuristics. An example is presented, showing how the method is applied to deconstruct a heuristic and identify individual separate criteria for evaluation. The same approach is later applied to the issues identified in the user test of Chapter 4 (Testing Heuristic Evaluation for Video Games). The terminology is then unified so that the same terms are used to describe issues and heuristics, which allows for a more specific and unambiguous approach to evaluation. Matching the same terminology between

issues and evaluation criteria makes the evaluation more transparent, objective, and improves the inspectability of the process. Ultimately the improvement in evaluator resources should also facilitate improvements in inter-rater reliability as well.

5.5.1 Content Analysis

> "Content analysis is defined as a research method for investigating problems by ...identifying characteristics of the message for the purpose of making inferences."
> Hsieh and Shannon (2005) cited in Cole (1988)

Content analysis has a long history of use, particularly in the humanities. It forms the basis of Grounded Theory, but that method goes further to develop a theory to describe the phenomena being analysed. As a qualitative method intended to work with a wide variety of material, content analysis is rather informal, relying on an interpretive analysis of the sources. The general approach is broadly defined by the following procedure,

- Source data is collected.
- The sources are reviewed to get an impression of the whole.
- Units of analysis are defined.
- Codes are iteratively created to describe the units being analysed.
- Codes are sorted into related categories.

The source data for the study presented in this chapter is the heuristic and issue text from Chapter 4 (Testing Heuristic Evaluation for Video Games). Units to be coded are usually words, phrases, or larger sections of text. In the procedure used for this chapter, the units of analysis are the individual words or phrases that refer to aspects of usability problems, and aspects of the game and the gaming experience. The categories used in the latter stages of the analysis are the principal components identified earlier in Section 4.4 (Validating Evaluation Themes).

The process is largely iterative and subjective, and the intention is not necessarily to produce a coding that other analysts can reproduce,

> "Because this process is difficult to describe and to communicate, qualitative studies tend to be carried out by analysts working alone, and replicability is generally of little concern."
> "Qualitative researchers tend to apply criteria other than reliability and validity in accepting research results."
> Krippendorff (2004)

The content analysis in this chapter uses heuristics (and later, reported user test issues) as the source material to be analysed and coded to identify potential terms for design and evaluation criteria.

5.5.2 Method

This section describes the process used, and takes a detailed look at an example heuristic to show how deconstructing its components can clarify differences in evaluator rating.

Specifically content analysis is used to decompose the heuristic into its constituent parts, coded as either usability outcomes or interaction breakdowns. Usability outcomes are empirically observable consequences as seen in user tests, defined as Efficiency, Effectiveness and Satisfaction. Interaction breakdowns are the the underlying causes of misunderstanding, erroneous action, physical or perceptual faults. Each analysis unit could potentially involve multiple outcomes and / or breakdowns.

All of the content analysis in this thesis was conducted by the author alone, and is not intended to represent a definitive or exhaustive deconstruction. The purpose is principally to help define a framework within the scope of this thesis. This analysis could be adapted, or other novel models could be constructed and used, and indeed this would be desirable as a way to extend the framework to address other games, genres, or experiences.

5.5.3 Identifying Breakdowns

Heuristics written in the form of *design principles* tend to imply positive usability outcomes which would constitute conformance to the heuristic criteria, and those written in the *analytical* form tend to explicitly state negative outcomes that constitute violation of the criteria. Interaction breakdowns are either explicitly stated, or assumed / inferred from the heuristic. In the positive form they define the requirements for heuristic conformance, which result in positive usability outcomes. In the negative form they are precursors which result in failure states with negative outcomes, and define the criteria for violation of the heuristic. In order to demonstrate these distinctions an example is presented in the next section.

5.5.4 Heuristic Content Analysis

Recall the first of the heuristics discussed earlier,

> Heuristic 136: "The skills needed to attain goals are taught early enough to play or use later, or right before the new skill is needed"

This example consists of the following relevant content analysis units:

- "skills needed"
- "attain goals"
- "taught"
- "early enough"
- "right before the new skill is needed"

Each of these are categorised in relation to the analytical framework of breakdowns and outcomes.

"skills needed" implies appropriate user actions to result in successful usability outcomes. As such it refers to a positive form of interaction, i.e., the absence of a breakdown.

"attain goals" refers to positive performance-based Effectiveness *outcomes* (e.g., completion rate) and possibly Efficiency in cases where goals are time-constrained, or where the degree of success is quantified by time.

"taught" suggests that the skills may not necessarily be obvious or intuitive, but may require teaching, learning and practice.

"early enough" implies that a breakdown in learning could occur, and that time is required to practice the skill.

"right before the new skill is needed" implies that breakdowns in recall could occur.

5.5.5 Content Analysis Isolates Evaluation Criteria

Referring back to the individual evaluators' ratings, this analysis can shed light on why they produced weak inter-rater reliability. Rather than identifying the cause of the problem, Evaluator 1 recognised a *design principle* that could have been used to prevent it from occurring. Their judgement was based on the assumption that a "skill" had to be "taught" in order to understand the game's displays. In contrast, Evaluator 2 recognised an alternative design principle which could have resolved the issue, but felt that as a negative usability *outcome* had not occurred, the "attain goals" criterion had not been violated. The ratings of Evaluator 1 and Evaluator 2 deal with assumed breakdowns in learning or recall, however the low ratings by Evaluator 3 do not accept this position, but rather come as a result of both rejecting the assumption that tuition was needed in the first place, and recognising that a negative outcome had not actually occurred. As such, none of the available heuristics adequately identified the *cause* of the problem.

5.6 Discussion

This detailed analysis has explained the decision making process of each evaluator, shown their particular biases that produced the evaluator effect, and revealed those areas which the available heuristics had not adequately addressed. Additionally it provides a method to analyse, critique and compare heuristics, and to understand why they produce different results. The high degree of specificity involved revealed a concrete cause of the issue, and the usability outcomes that it can produce.

5.6.1 Composite Heuristics Reduce Inspectability and Reliability

The original evaluation teams of the six heuristic sets used in Chapter 4 (Testing Heuristic Evaluation for Video Games) achieved agreement in their own studies through private discussion during evaluation. Without the decisions made during those discussions being instantiated as

formal, objective evaluation processes in the methodology, repeatability and validation of their results is not possible.

Furthermore, choosing just a single heuristic per issue can contribute to problems where differences in evaluators' interpretations result in different decisions as to which single heuristic best explains each issue.

5.6.2 Proposing The Playthrough Evaluation Framework

In order to resolve the problems with reliability seen in Chapter 4 (Testing Heuristic Evaluation for Video Games), this thesis proposes the playthrough evaluation framework as a way to reconcile the gap between general heuristics and specific issues by creating a reliable, hierarchical structure with a focus that's more relevant for first-person shooter games, and which benefits from design and evaluation knowledge specific to the domain.

As was shown in Chapter 4 (Testing Heuristic Evaluation for Video Games), it can often be reasonable to use a variety of different heuristics to explain most usability issues in this complex domain. The problem is that heuristics operate at a high level that tends to concentrate on player experience outcomes rather than usability breakdowns. As such, any particular breakdown can potentially result in a number of different outcomes, often apparently unrelated to one another.

For example, consider Issue 40: "Moderator gives tutorial on how to use the alien". In this case the player is having some difficulty and so the moderator intervenes to provide help. This could be considered as a violation of heuristics to do with tutorials, learning skills, controls, error prevention, etc. However, the perspective provided by the player action framework acknowledges not only the observable outcomes, but also provides scope for the evaluator to document and explore potential causal relationships. The candidate codes to consider would include events indicating when the skill was first introduced, needed, and how it was (mis)used by the player. At this level of analysis reliability is expected to be higher, as evaluators will normally be able to agree on such concrete and specific observations. From there evaluators may speculate about or infer possible causes. At this level reliability is much more subject to the evaluator effect.

5.6.3 Playthrough Evaluation

The main purpose of the playthrough evaluation framework presented in Chapter 6 (The Playthrough Evaluation Framework) is to define a novel methodology, playthrough evaluation, to reliably code empirical observations from user test sessions. By providing an initial, reliable base of observational data, evaluators start with a sound platform from which to form their interpretations. In this coding scheme, empirical observations are explicitly recorded. This makes it much easier to identify where and why problems with inter-evaluator reliability are introduced. In addition, this also facilitates a reflective critique of the method itself, by exposing weakness in the method that contribute to reliability problems. Furthermore it suggests possibilities for appraising evaluators' ability to apply the method by comparing their coding against established benchmarks or reference standards.

5.6.3.1 Playthrough Evaluation Maps Issue Space in More Detail

In discussing Usability Inspection Methods (UIMs), Cockton et al. (2004) state,

> "We must ensure that a false positive is not due to a flaw in method assessment. Similarly, an unpredicted problem must also be shown to be due to the UIM and not to the assessment."

In the case of playthrough evaluation, the more detailed structure is particularly amenable to these concerns. The evaluator's justification for their decisions is explicitly documented, and can be analysed in detail. Any deficiencies in the evaluation that are caused by the method itself can be observed and corrected, passing on the benefits of this introspection to future researchers. Cockton et al. (2004) also argue that the assessment of usability evaluation methods must include the identification of:

> "true and false positives, true and false negatives, and oversights, by analysing predictions from inspection against problems discovered in user testing."

This position is also well supported by the playthrough evaluation framework. The framework has provision for use as a documentation tool for user test sessions, where the explicit criteria structure facilitates introspection, analysis, and critique.

5.7 Conclusion

Evaluators were interviewed and asked to talk about how they rated issue with the heuristics in Chapter 4 (Testing Heuristic Evaluation for Video Games). By exploring their interpretation of the heuristics and issues, and reflecting on the low inter-rater reliability in their ratings from a sample issue, three different styles of heuristic were identified: those based on interaction breakdowns, usability outcomes, and general design principles. Content analysis was used to decompose an example heuristic into its constituent parts, and analysed through a framework of interaction breakdowns between user and system, and subsequent usability outcomes. The explicit separation of interaction breakdown and usability outcome criteria provided insights into why the evaluator effect occurs for individual heuristics. Evaluators in the study were using different criteria from each heuristic and issue to determine which rating to assign. In effect they were evaluating different components of the system and interaction, which resulted in low inter-rater reliability.

While the heuristic evaluation in Chapter 4 (Testing Heuristic Evaluation for Video Games) exhibited weak inter-rater reliability, the principal components analysis results from Section 4.4 (Validating Evaluation Themes) suggested a core set of themes that the heuristic sets in the literature address. Although the specific phrasing of individual heuristics is subject to misinterpretation by evaluators, they do still say something useful about game evaluation. How they're presented and used, though, could be improved. As was shown earlier, many heuristics are phrased in general or uncontroversial ways which have little bearing on evaluation and improvement. In the Discovery and Analysis REsources (DARe) terminology of Cockton, Woolrych, Hall, et al. (2003), these heuristics offer relatively weak evaluator resources in the way of problem

Discovery and Analysis. This is due to their high degree of abstraction, which makes heuristics useful as an indicative guide or reminder for designers and expert reviews, but which produces ambiguous and unreliable results when they are used for more precise evaluation.

Chapter 6 (The Playthrough Evaluation Framework) systematically applies the unpacking technique described in Section 5.4 (Heuristic Types) to each of the heuristic candidate areas identified earlier in Section 4.4 (Validating Evaluation Themes). This derivation of novel resources for discovery and analysis provides detailed and objective criteria with which to evaluate each aspect of the game, and to assist in the categorisation of issue breakdowns and outcomes. Chapter 6 (The Playthrough Evaluation Framework) goes on to show how these resources are used to conduct and analyse a playthrough evaluation. Chapter 7 (Testing Playthrough Evaluation) empirically tests the method in comparison to heuristic evaluation, and shows substantial improvements to inter-evaluator reliability.

Chapter 6

The Playthrough Evaluation Framework

6.1 Introduction

This chapter presents the playthrough evaluation framework that systematically adapts the design and evaluation knowledge of heuristics into a more structured form, and which uses the playthrough evaluation method for more reliable usability evaluation. The previous chapters are reviewed in order to summarise the motivation for this new framework and evaluation method. Related literature is reiterated from Chapter 2 (Literature Review), showing how the playthrough evaluation framework developed from prior work. The steps taken to develop the framework itself are outlined, and in particular the analytical method used to derive the coding scheme.

The chapter begins by identifying representative heuristics for each of the components derived in Chapter 4 (Testing Heuristic Evaluation for Video Games). These heuristics are unpacked with the content analysis method introduced in Chapter 5 (Exploring Evaluation Resource Specificity), giving a set of key criteria to evaluate each component in a more specific and concrete way than just the heuristics alone. Representative issues for each heuristic are then identified from the rating data of Chapter 4 (Testing Heuristic Evaluation for Video Games), and unpacked using the same content analysis approach. The key terms extracted from all of these heuristics and issues are then aggregated into a single set of "events" that describe the potential breakdowns, outcomes, and design features to be evaluated. Use-case scenarios are constructed for each component, consisting of sequences of events from the component's heuristics and issues. Evaluators use these scenarios as a checklist of criteria to evaluate in the playthrough evaluation methodology.

This chapter the describes how the playthrough evaluation method is applied, and is followed by Chapter 7 (Testing Playthrough Evaluation) which empirically tests the inter-evaluator reliability of playthrough evaluation and compares it against a benchmark of heuristic evaluation.

6.2 Background

The quantitative analysis in Chapter 4 (Testing Heuristic Evaluation for Video Games) demonstrated that evaluators rate heuristics very differently to one another, resulting in poor inter-rater reliability. Chapter 5 (Exploring Evaluation Resource Specificity) presented a qualitative exploration which discussed some of the reasons for this, including the subjectivity of interpreting heuristics and the poor availability of resources for issue discovery and analysis. As the most common and well developed method for evaluating video game usability, heuristic evaluation does provide useful domain-specific high level design and evaluation guidelines. However, due to their level of abstraction a large degree of subjectivity is involved in interpretation, and subsequently poor inter-evaluator reliability results. Despite evaluators rating individual heuristics very differently to one another, the principal components analysis in Chapter 5 (Exploring Evaluation Resource Specificity) demonstrated that evaluators treated *related* heuristics similarly. This suggested that there are a core set of heuristics that all deal with the same areas. The weak inter-rater reliability seen in heuristic evaluation is related to the inherent subjectivity in the method, and the ambiguity of relating simple heuristics to complex issues, as evaluators interpret the heuristics and issues differently to one another. In order to improve the reliability of usability evaluation, clearer guidance is needed for how to interpret and use the heuristics' design and evaluation resources in the identified core areas.

In this chapter content analysis is applied to decompose the heuristics and issues from Chapter 4 (Testing Heuristic Evaluation for Video Games) into their constituent analysis units which are much more precise and concrete than the composite heuristics they constitute. These detailed criteria are used in playthrough evaluation to make the process much more specific, measurable, and less ambiguous than the original heuristics.

6.2.1 Defining Issues

6.2.1.1 Issues Are Complex and Specific, Heuristics Are Simplified and General

In the real world, issues are complicated and multi-faceted, involving the interaction of many different design components and user faculties. Heuristics, on the other hand, tend towards abstraction and are intended to be more general design guidelines that can apply in a wide range of circumstances. The consequence of these differences in specificity is that a single heuristic is unlikely to completely explain any given issue, though may well explain a part of it. Concomitantly, any single issue is likely to have different aspects of it explained by several different heuristics.

6.2.1.2 Locating Problems in Issue Space

The playthrough evaluation framework derived in this chapter addresses the disconnection between the complexity of issues and the abstract generality of heuristics used to evaluate them. It approaches this task by firstly acknowledging that issues typically involve a nexus of inter-related design issues, user experiences, and usability consequences. It proposes that the components involved can be identified and evaluated with a greater specificity than is provided

by general, abstract heuristics.

A spatial metaphor may help to illustrate the point. A heuristic can be thought of as a simplified window into a multi-dimensional space, just like a camera provides a flattened 2 dimensional view of a 3 dimensional space. We can see whether an object is inside the view of the camera of not, and can likewise rate whether a heuristic is an appropriate description of a usability issue or not. Different cameras can be be oriented to look into the space from different perspectives, and some will view the object in question while others will not. Similarly, different heuristics can be used to look into the "space" of usability issues. Some heuristics are entirely irrelevant to some issues, so we could say that the issue is not within the issue-space view projected by the heuristic. For example, an issue about the control scheme is very unlikely to be addressed by a heuristic dealing with the realism of the game's audio.

In the ideal case, each heuristic would offers a completely orthogonal perspective so that all parts of all possible issues would be entirely and unambiguously inside the view projected by one and only one heuristic window. The heuristics described in the literature often try to achieve this by partitioning them into separate categories. Malone (1982) for example used the informal categories "Challenge", "Fantasy", and "Curiosity" to divide the space of intrinsic motivation. Principal components analysis is a sound statistical approach for revealing the underlying themes represented by many different heuristics, as described by Nielsen (1994a) to derive his canonical ten heuristics, and similarly followed in Chapter 5 (Exploring Evaluation Resource Specificity). This method is particularly suited to expose orthogonal components, so produces a good map of the space.

Nonetheless, heuristics collapse the multi-dimensionality of real issues into a one-dimensional evaluation of whether a complex, multi-faceted issue is addressed by a simplistic heuristic or not. In other words, whether the issue is inside the view projected by the heuristic window into the issue space.

Nielsen's rating scale provides a 5 point linear dimension that represents how well an issue is described by a heuristic. In other words, how well the issue fits inside the view projected by the heuristic window. However, this is no guarantee of orthogonality, and there is nothing to stop multiple heuristics from projecting viewpoints that both completely include the space occupied by a single issue. This is seen especially with the more abstract / reflective kinds of heuristics that Nielsen produced. For all issues in Chapter 4 (Testing Heuristic Evaluation for Video Games), Nielsen's heuristics were rated with very low scores by all three evaluators. That is, each heuristic window only partially projected onto a portion of the issue, and didn't provide an unambiguously clear view of it (i.e., explanation of why it is a problem).

This topic is picked up by Doubleday et al. (1997) which discusses some of the problems involved in an evaluation,

> "Boundaries between 'errors' can be fuzzy and overlapping; one person's stated error may comprise several symptoms reported by others as separate errors."

The more ambiguous the heuristics are, the more ambiguous and blurry is the viewpoint they project into the issue space. Furthermore, despite using sound statistical approaches to divide the space as orthogonally as possible, in any case of dimensional reduction like this

there is an inevitable loss of data. The borders between heuristics are ambiguous, porous, and a further source of disagreement amongst evaluators.

While heuristics describe idealised abstract cases, the space around them is not well illuminated. Inevitably issues in real world usability evaluations are complex, and resist simplistic mapping to this kind of idealised representation. In turn this traditionally necessitates inter-evaluator discussions to informally explore the design space surrounding the ideal heuristic case, in order to determine the extent to which a real issue exists within the space projected by the heuristic.

Novices Benefit from Explicit Guidance

Expert evaluators have been exposed to many complex, ambiguous, real world issues, and so have had plenty of opportunity to consider the relationship between issue and heuristic space. As such they are in a stronger position to be able to judge whether an issue is relevant to a heuristic or not. Novice evaluators have not had as much practical experience exploring and reflecting on the heuristic space, so are more likely to have blind spots in their evaluation. Heuristic evaluation is best conducted by expert evaluators in formative cases, but a usability evaluation method designed for summative evaluations by novices, such as playthrough evaluation, should provide more clarity in its methodology.

Structuring Design and Analysis Resources

Heuristics are usually organised into flat arbitrary categories (e.g., mechanics, usability, curiosity, fantasy, etc.). These categories are convenient for conceptual organisational purposes, but little has been studied into the effect they have in terms of discovery and analysis resources. Heuristics tend to be composites of criteria, expressed at various different conceptual levels, and with varying degrees of analytical criteria for discovery and analysis. Similarly, the categories they are organised into are sometimes more to do with outcomes, sometimes more Breakdowns, sometimes different components of the game (such as head-up display, controls), and sometimes they are more to do with the game as vehicle for play (e.g., aspects of player experience such as aesthetics, challenge, etc.)

In order to operationalise the design and evaluation knowledge implied in the heuristics from the literature, this chapter redefines heuristics into their component criteria, and reorganises them in a hierarchy. At the top level are the core components identified by the principal components analysis in Chapter 4 (Testing Heuristic Evaluation for Video Games). e.g., Skills; Challenge; UI; Controls. This level is a convenient starting point for evaluators when they first notice a problem, but perhaps before they are able to identify the underlying cause of the breakdown. Each component shows the associated heuristics, along with representative interaction scenarios, listing the derived criteria that are used to evaluate an issue in this area.

6.2.2 Related Literature

6.2.2.1 Interaction Theory

Norman's theory of action (Norman, 1986) is widely used as a theoretical underpinning for a number of other evaluation frameworks (Baauw et al., 2005; Polson and Lewis, 1990; Polson et al., 1992; Springett, 1998; Winter et al., 2008) and provides the backbone to playthrough evaluation framework as well.

The theory separates user-system interaction into seven stages, and recognises that problems can occur within each:

1. Forming a goal.
2. Forming an intention.
3. Specifying an action.
4. Executing the action.
5. Perceiving the system state.
6. Interpreting the system state.
7. Evaluating the outcome.

This separation, much like the distinction between breakdowns and outcomes as seen in Lavery et al. (1997), facilitates detailed analysis of usability issues. Indeed, the example breakdowns in Lavery et al. (1997) show a clear influence from Norman's seven stages of action,

- the user forming an inappropriate goal.
- the user selecting an inappropriate action.
- the user not perceiving the feedback.
- the user misinterpreting the feedback.

This explicit separation and enumeration of different types of breakdown provides detailed resources that can be helpful for the analysis of usability problems. The theory itself does not define a method or other concrete procedures for evaluation, however it has been widely influential in the development of practical tools, such as the user action framework.

6.2.2.2 The User Action Framework Provides Detailed Analysis Resources

The user action framework is a comprehensive taxonomy of usability issues based on Norman's theory of action (Andre, 2000; Andre et al., 2000, 2001, 2003; Capra, 2001; Catanzaro, 2005; Hartson et al., 1999; Keenan et al., 1999; Mahajan, 2003; Mentis and Gay, 2003; Sridharan, 2001). It provides a hierarchical structure for the categorisation of breakdowns with general usability issues. This offers much in the way of analysis resources, but little for problem discovery. It is well suited to analysis of the underlying breakdowns in actual or potential issues, when they have been already identified either through user testing for actual issues, or expert inspection in the case of potential issues.

The framework is used during evaluation by considering each stage of the interaction cycle for each task involved, and checking the system against numerous potential problems that could occur. For example, a task might involve the user pressing a button to invoke a particular function and then parsing the system's feedback. In this case the evaluator would first consider the initial branch of the hierarchy, "Planning", that describes potential issues in the user's overall understanding of the available system functions. The branch of the hierarchy would be examined, and potential or actual cases identified where the user's task planning could go wrong. Types of problem identified in this stage could involve the user's lack of awareness about the necessary system functionality needed to achieve their goal, for example. Following this the analyst would then consider the branch dealing with the use of the interface to achieve the goal planned. While the user might understand that the system can be used to achieve their goal, they might not notice the necessary design features, or understand how to use the interface to perform them. Further branches in the hierarchy describe potential problems of physically using the interface, and understanding the system's response.

Reliability is reported as strong for the topmost levels of the hierarchy, but decreases at the more specific lower levels of the tree (Andre et al., 2001). The lowest levels of detail in particular proved to be difficult for evaluators to agree on, with Cohen's Kappa values being similarly low to those obtained by heuristic evaluation (0.325). This is a reasonable example of a real evaluator effect, where the methodology is fully specified, and yet where evaluators still have different opinions about the most likely cause of problems.

The user action framework vastly expands on Norman's seven stage model by defining hundreds of nodes in the hierarchy. This level of detail could be suitable for task-oriented evaluation, especially where a great deal of reflection is needed about every part of the interaction cycle, such as with cognitive walkthrough. For the purposes of this thesis, such an extreme level of detail is excessive. Defining tasks at this level of detail is usually not feasible with first-person shooter games, partly due to the dynamic nature of interaction which produces different paths through the system for each session, and partly due to the increased degree of complexity compared to traditional domains. The user action framework is also limited to cognitive and physical actions for more traditional Windows, Icons, Mouse, Pointer interfaces, so would need to be adapted to address the specific characteristics of video games. Norman's theory was originally developed for the field of *Cognitive Engineering*, which explains its emphasis on knowledge resources and structures to support the user. However, in the user action framework studies, evaluators found the cognitive language difficult to understand, which is likely to have contributed further to deficiencies of inter-rater reliability.

The user action framework is used in this chapter to help derive the playthrough evaluation framework. Content analysis is applied to the issues and heuristics from Chapter 4 (Testing Heuristic Evaluation for Video Games), and nodes from the user action framework hierarchy help to categorise the analysis units identified. The comprehensive structure provided by the user action framework hierarchy helps to ensure that the scope of categories derived is broad enough to address a wide range of specific usability problems. The user action framework proved difficult for evaluators to use, as it specifies problems using specialist cognitive terminology. The categories derived by the playthrough evaluation framework resolve this difficulty

by using terminology that's more relevant to the domain of game evaluation.

6.2.3 Defining Interaction Scenarios

Scenario-based design is an approach to usability evaluation that involves describing a scenario of potential user activity with a system, and has been used in the literature to derive usability specifications (Carroll, 1995; M. B. Rosson and J. M. Carroll, 2003). These are defined in terms of textual *scenarios* describing typical user activities, and *claims* hypothesising usability outcomes that could result from particular design decisions referred to in a scenario. During claims analysis an expert assesses these specific criteria referred to by the scenarios for their potential impact on the usability of the system.

This chapter builds on the approach of scenario-based design and claims analysis, but applies some novel modifications for this new evaluation context. Scenario-based design was originally intended for formative evaluation while the product is still under development, and so the scenarios are imagined narratives describing how a user might interact with a hypothetical system. Instead, with summative evaluation in the playthrough evaluation framework the game already exists in a playable form, and so scenarios can be created that represent the system as it actually is, rather than how it is envisioned to be in the future.

Similarly, traditional claims analysis applies a form of content analysis by reflecting on the hypothetical scenarios and identifying key words or phrases that indicate potential usability outcomes. However, rather than having to make do with analysis of hypothetical scenarios, the playthrough evaluation framework takes the object of its analysis to be actual user test problems reported in Chapter 4 (Testing Heuristic Evaluation for Video Games).

Content analysis of the issues is used to define the key terms involved in the scenario. Heuristics are similarly analysed for the key terms that can be used to evaluate the scenario. Claims identified during these analyses are categorised with the help of the detailed structure from the user action framework. This approach is similar to that described in Somervell (2004) which also adapted scenario-based design using traditional claims analysis, categorising usability problems with a simplified user action framework-like tree. The purpose of that study was to derive novel heuristics from the claims identified in the scenarios. The analytical stage of playthrough evaluation framework however does not generate further heuristics, as the heuristic evaluation method itself has been shown to be a source of poor reliability. Instead, a novel coding scheme is derived from these analyses, which is used to describe user test sessions and evaluation criteria in a common terminology. This coding scheme is then used as an improved problem discovery and analysis resource. Chapter 7 (Testing Playthrough Evaluation) demonstrates the improvement in reliability that playthrough evaluation offers over traditional heuristic evaluation.

6.3 Deriving the Framework

6.3.1 Decomposing Heuristics

The heuristics from the literature were analysed in terms of Breakdowns (informed by the user action framework), usability Outcomes (Effectiveness, Efficiency, and Satisfaction), and game

components (e.g., controls, goals, skills, head-up display, etc.)

The purpose was to facilitate three different entry points to evaluation:

1. Problem discovery (by observing usability outcomes).
2. Problem prediction (by systematic consideration of game components).
3. Problem analysis (by interpreting breakdown causes).

6.3.2 Method

For each of the areas identified in the principal components analysis the subset of heuristics involved were considered. Each of the heuristics was unpacked into analysis units. From these decompositions a criteria tree was constructed that relates all of the partial heuristic criteria to dependent breakdowns and resulting outcomes.

6.3.3 Identifying Representative Heuristics for Decomposition

The heuristic sets used in Chapter 4 (Testing Heuristic Evaluation for Video Games) were considered for inclusion. Nielsen's set which is primarily of the abstract / reflective type was rejected, and only those sets that exhibited characteristics of design principles or critical analysis were retained. The purpose was to take the implicit knowledge contained in the heuristics, and extract clear explicit criteria which can be reliably used to confirm violation or conformance to the heuristic. Abstract / reflective sets such as Nielsen's original 10 are considered to be more useful for prompting the implicit, subjective, experience-based understanding that an expert already has, but which is not externally defined elsewhere.

The heuristics used are as follows:

1. Federoff (2002)
2. PLAY Desurvire and Wiberg (2009)
3. Pinelle et al. (2008a)
4. GAP Desurvire and Wiberg (2010)
5. Korhonen et al. (2009) (excluding mobile and multiplayer components)

Heuristics in these sets exhibit some aspects of each type identified in Chapter 4 (Testing Heuristic Evaluation for Video Games): design principle; abstract / reflective; outcome analytic. The latter are easiest to decompose as they are the most specific, though there is useful design knowledge contained in the other forms too. In order to extract this knowledge it is useful to consider real world issues that were rated highly as violating these heuristics. Following the analysis of the heuristics, a similar process was applied to the issues from Chapter 4 (Testing Heuristic Evaluation for Video Games) too.

6.3.4 Identifying Decomposition Candidates

Representative heuristics for analysis were selected for each component. The following criteria were used to identify candidate heuristics and issues for each of the principal components:

- Cases where at least one evaluator used a rating of level 5 ("Complete explanation of why this is a problem")
- Cases where variance between the three evaluators was 5 or greater (in the range of 0 to 8.33).

These candidates represented cases where heuristics were considered by at least one evaluator as being important for describing the issue, but where they may not be a consensus amongst all of the evaluators. A similar procedure was later applied to each issue.

The intention was to identify strong candidates for content analysis and to unpack them in terms of outcomes, breakdowns, and other events in the game-player interaction. The end product is a database of relationships between events, breakdowns, outcomes, heuristics and representative issues.

6.3.5 Heuristic Decomposition Template

Heuristics were decomposed using the following template.

- ID.
  A unique identifier for this heuristic.
- Heuristic Source.
  The original publication of the heuristic.
- Heuristic Text.
  The heuristic itself, usually only one or two lines.
- Analysis units.
  Words or phrases from the heuristic that are the key terms used to derive evaluation criteria.
- Outcomes.
  Analysis units that refer to usability outcomes.
  - Effectiveness.
  - Efficiency.
  - Satisfaction.
- Breakdowns.
  Analysis units that refer to breakdowns where interaction did not proceed in the way expected by the player or designer. The user action framework was used to identify specific usability issues that each heuristic suggested.

6.3.6 Example Heuristic Decomposition

This section presents an example of a heuristic decomposed into outcomes (effectiveness, efficiency, satisfaction) and breakdowns (derived from the user action framework tree hierarchy), and referring to items of the game involved (e.g., goals, skills, HUD, AI, controls, etc.)

- ID: 94
- Source: PlAY-ng (Desurvire and Wiberg, 2009)
- Heuristic 94: "Status score Indicators are seamless, obvious, available and do not interfere with game play"
- Analysis units:
  - "Status score Indicators"
  - "seamless"
  - "obvious"
  - "available"
  - "interfere"
  - "game play"
- Outcome:
  - "Status score Indicators do not interfere with (Effective) game play."
  - "Status score Indicators do not interfere with (Efficient) game play."
  - "Status score Indicators do not interfere with (Satisfying) game play." (e.g, frustration, annoyance).
- Breakdown:
  - "Status score Indicators are seamless"
  - "Status score Indicators are obvious"
  - "Status score Indicators are available"

6.3.6.1 Defining Domain Specific Analysis Units

The heuristic does not explicitly define what a "Status score Indicator" is, so each heuristic evaluator would normally be responsible for interpreting this individually. As shown in Chapter 4 (Testing Heuristic Evaluation for Video Games) this can potentially be a source of problems with respect to reliability. For example, the term "indicator" could be interpreted by one evaluator as only meaning text or numeric information displays that are always shown as part of the head-up display, such as the player character's health count. Alternatively, another evaluator might interpret this to mean any form of indication, such as a red border to the screen whenever the player character receives damage. What's more, some games do away with a classic head-up display and indicate status with in-game artwork. For example, the available ammunition in *Aliens Vs. Predator* may be displayed on the in-game weapon itself. This may, coincidentally also be an example "seamlessness".

Interpreting this heuristic is further complicated by the lack of definition regarding the term "Status score". It's unclear whether this was intended to only refer to a literal score, or perhaps any kind of status *or* score, as in "status/score".

The intention of the playthrough evaluation framework is to provide more domain-specific structure and guidance for the evaluator. Following the distinction made by the user action framework, two similar classes are defined for the playthrough evaluation framework:

- "UI/art/indicator"
- "feedback"

The distinction between the two is that "feedback" only occurs in response to interaction, whereas the "UI/art/indicator" class does not.

Feedback could include visual, auditory, haptic, or potentially other modalities. Feedback only occurs in response to an interaction, so is not always a fixed part of the game's output like an indicator.

In this terminology, "indicators" means typical information displays that are fixed, such as a counter to show the player character's health or ammunition, or a minimal. This class additionally includes readouts on a traditional User interface or Head-up display, displayed as part of the game's artwork.

Deriving Categories

Following the content analysis of the heuristics, outcomes and breakdowns were aggregated across the set. The next stage applied the same approach to the analysis of the issues that were rated highly for each heuristic. Once all issues had been analysed the outcomes and breakdowns were again aggregated with the same items from the heuristics. Following the procedure for content analysis, related codes were merged into categories. These categories were then reformatted into a novel form as events, and structured into scenarios describing the general sequence of events to be evaluated for each component.

Following the identification of codes for each item, categories are defined that could be used during a practical evaluation.

For example, Heuristic 94: "Status score Indicators are seamless, obvious, available and do not interfere with game play",

"Status score Indicators are seamless":

The concept of "seamlessness" would need to be better defined in terms of usability potential. It's unclear whether this is an aesthetic quality that would only affect user satisfaction in the sense of the indicator fitting with the visual theme of the display. Alternatively it may imply that a non-seamless indicator could also contribute to usability outcomes of effectiveness or efficiency as well.

To address the "availability" of the indicator in the category "Status score Indicators are available", two further events are derived dealing separately with static indicators and dynamic feedback,

- Event 28: (UN/necessary/desirable) UI/art/indicator (IS/NOT) visible
- Event 29: (UN/necessary/desirable/expected) feedback (DOES/NOT) occur

These events address with objective existence and *potential* visibility. Whether the player actually notices and understands them depends on the more subjective notion of "obviousness", dealing with the user's ability to notice and understand an indicator,

The breakdown dealing with the obviousness of the status score indicator is addressed by the following events in the playthrough evaluation framework,

- Event 26: Player (DOES/NOT) notice necessary/desirable UI/art/indicator
- Event 27: Player (DOES/NOT) understand/recognise purpose/meaning of UI/art/indicator
- Event 30: Player (DOES/NOT) notice/recognise necessary/desirable/expected feedback

- Event 31: Player (DOES/NOT) understand feedback

6.3.7 Real Issues Flesh out Heuristics

It is useful to describe possible scenarios, or actual case studies that could lead to violation, particularly when heuristics provide little in the way of evaluation resources. By considering real world examples we are able to start investigating the specific breakdowns that could lead to usability outcomes, and so enable the heuristic to be used in a more analytical and reliable manner.

6.3.8 Identifying Representative Issues for Decomposition

Each of the issues from Chapter 4 (Testing Heuristic Evaluation for Video Games) were considered. In cases where at least one evaluator rated a heuristic at a high level, the issue was analysed for potential criteria matching. This process was conducted in an iterative way, beginning with issues rated at the highest, level 5 (Complete explanation of why this is a problem), then proceeding to lower levels, concluding at level 3 ("Explains a major part of the problem, but there are some aspects of the problem that are not explained") or greater. Content analysis was again used to identify analysis units in the issue text, and events were derived to represent individual criteria involved.

6.3.9 Defining Interaction Scenarios

Having derived criteria by decomposing the heuristics, a coding scheme for documenting representative interaction scenarios is defined. The codes are composed from the criteria derived through the content analysis. This means that the terminology used to represent the scenarios and the language used to evaluate the session is the same.

An example issue shows how the events from the heuristic mentioned earlier can be used for evaluation,

Issue 13: "Player comments almost immediately that 'There's this thing at the bottom showing me which way to go.'".

This issue describes a case where the player has correctly noticed and understood the purpose of the indicator, and conformance to the heuristics is evaluated using the events listed above. i.e.,

- "Necessary/desirable indicator (waypoint) IS visible."
- "Player DOES notice Necessary/desirable indicator (waypoint)."
- "Player DOES understand/recognise purpose/meaning (navigation) of indicator (waypoint)."

Existing approaches to user test analysis and usability evaluation rarely define common formal coding schemes, and instead employ informal evaluation methods. The consequence of this is ambiguity and poor reliability in evaluation results. By linking the two steps of analysis and evaluation it not only has the potential to improve reliability, but also makes the methodology explicitly exposed for critique. In the case of a traditional heuristic evaluation being validated against user test, it is not possible to identify whether the poor reliability is

due to incorrect analysis of validating data (i.e., identifying what has occurred in the test session), incorrect matching of actual test data and predictions, or simply individual evaluator performance.

In the novel approach developed here, the criteria and process of evaluation is exposed for critique, and hence for improvement and testing.

6.3.10 The Playthrough Scenario

For each component an interaction scenario was constructed using the events identified from the associated heuristics and issues. Similar to Matera's Abstract Tasks (Matera, 1999), the scenarios represent a template for evaluation, indicating the expected sequence of interaction events which helps evaluators identify deviations and errors. Each of the events were categorised according to stage in the interaction scenario, structured according to three overall types: context; breakdown; outcome.

Consider an example of problem where a player fails to use the right control and so fails to complete a task. This scenario would be structured as follows,

- Context:
  Contributing factors or design aspects that establish the scenario, but which do not naturally fit into the breakdown/ outcome categories; events preceding a breakdown but which predicate it. For example, a task was set that required the use of a certain control.
- Breakdown:
  When things went wrong. For example, the player using the controls incorrectly.
- Outcome:
  The problematic consequence of the breakdown. For example, the player may fail the level due to incorrectly using the controls.

This scenario could be transcribed using the playthrough evaluation events as follows:

- Context:
  - Event 18: New goal/task set (IMPLICITLY/EXPLICITLY)
  - Event 6: Skill/control/action/mechanic/feature/tactic (UN/necessary/desirable/expected) for goal/task
- Breakdown:
  - Event 23: Player (DOES/NOT) understand how to use control/feature/skill/mechanic
  - Event 8: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic used (UN/SUCCESSFULLY)
- Outcome:
  - Event 14: Effectiveness: Task (SUCCEEDED/FAILED, IN/COMPLETE)

6.3.10.1 Scenarios Provide Multiple Points for Issue Detection

A user test issue can be detected and analysed by a variety of different initial observations, at different stages of an interaction scenario. Consider for example, Issue 6: "Player sees the waypoint on the roof. Mashes buttons, but can't work out how to interact with it".

This report does not describe the actual problem in itself, though this can be inferred through some prior understanding of the game. In this scenario, the player has previously been

introduced to the waypoint navigation system, but has clearly not understood how it works. The preliminary breakdown, then, is in the initial exposure and training about the navigation system. It was expected that the player would understand the instructions and be able to use the waypoints without further problems. Clearly this was not the case for this player, and so the earlier breakdown manifested as a problem at this later point. Note that it may not have been possible to identify the breakdown earlier when the instructions were first presented to the player if there was no action required to demonstrate competence or understanding of them immediately.

This is a good case for systematic, structured evaluation. At this initial event (instruction presented), the evaluator would consider the principal components for relevant areas, such as,

- Component 1: Learning Skills (Controls, Mechanics, Tactics)
- Component 3: Manual & Tutorial (Help and Documentation)
- Component 12: Visual Representation Form & Function
- Component 14: Clear Goals

For each of these areas, representative interaction scenarios are examined for potential problems that could occur later in a play session.

The event that starts the scenario is,

- Event 1: Skill/control/action/mechanic/feature/tactic (IS/NOT) introduced

And the event that is expected to follow is,

- Event 4: Player (DOES/NOT) practice/demonstrate necessary/desirable/appropriate/expected/correct competence with Skill/control/action/mechanic/feature/tactic

Evaluation of this event will be straightforward if the feature is simple enough, or confirmed by observation through player utterance, or non-verbal behaviour such as performance testing. At a later point in the game, the player will presumably encounter the rest of the scenario pattern,

- Event 6: Skill/control/action/mechanic/feature/tactic (UN/necessary/desirable/expected) for goal/task
- Event 8: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/-tactic used (UN/SUCCESSFULLY)

The interaction scenarios for different components often share some of the same events. For example, "Event 6: Skill/control/action/mechanic/feature/tactic (UN/necessary/desirable/-expected) for goal/task" is used in the scenarios for "Component 1: Learning Skills (Controls, Mechanics, Tactics)" as well as "Component 2: Challenge". This helps to broaden the evaluation scope beyond the single component first identified by the evaluator. Unlike heuristic evaluation where evaluators typically choose a single heuristic to represent an issue, the scenarios in playthrough evaluation include several events that are related to and may have an impact on the issue in question. This means that even if evaluators were to start their evaluation using

different components, the interaction scenarios would guide them to consider some of the same key events.

Playthrough evaluation is conducted according to the procedure presented in Appendix C.1 (Playthough Evaluation Procedure). An explanation of the process is presented in the following section.

6.4 Performing Playthrough Evaluation

Evaluation begins with the participant playing the game in a natural way, without consideration to evaluation criteria. Footage of the game and the player are recorded simultaneously. Following the play session the footage is reviewed and used as the data to evaluate. The evaluator pauses the video whenever the evaluator identifies a problem that they experienced or that another player might have difficulty with. Problem detection is likely to be triggered by noticing evidence of a usability Outcome (such as task failure), or a possible design fault.

A new issue report is filed with a timestamp for the issue, a brief description of the problem, and a list of the relevant components.

At this stage the evaluator consults the player action framework (Appendix B.2 - Player Action Framework Tree). Each candidate component is considered, along with its related heuristics. If these heuristics appear relevant, then the evaluator reviews the event codes in the interaction scenario patterns for that component. Each event in the scenario defines a criterion for evaluation, and the evaluator records on the issue report whether the event was violated or not.

For example, looking at the components implicated by Issue 63: "Player comments that getting in to the vent was difficult. A few seconds later he comments again that he can't get down a hole" the evaluator would identify that controls and skills need to be included in their analysis. Candidate components that could be relevant include the following:

- Component 4: Usable Controls
- Component 11: Player in Control
- Component 15: Unreasonable/Unexpected/Unacceptable Errors (Error Prevention)

6.4.1 Example Playthrough Evaluation

Consider the case of an evaluator observing the following issue, Issue 19: "Player comments that 'I can't interact with this.' Others players have had the same problem. He needs to get the battery first"

Upon noting the observation, the evaluator considers the principal components and identifies the component dealing with controls as a candidate. The associated heuristic and interaction scenario are shown following,

- Component 4: Usable Controls.
  - Heuristic 23: "Controls should be intuitive and mapped in a natural way"
  - Context:

    * Event 24: Control/feature/skill/mechanic (DOES/NOT) default/conform to industry standard
  - Breakdown:
    * Event 22: Player (DOES/NOT) understand function/purpose/effect/consequence of feature/skill/control/mechanic
    * Event 23: Player (DOES/NOT) understand how to use control/feature/skill/mechanic
    * Event 7: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic use (IS/NOT) attempted
    * Event 8: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic used (UN/SUCCESSFULLY)
  - Outcome:
    * Event 16: (DIS/SATISFACTION)
    * Event 15: Efficiency: Resource expenditure too (HIGH/LOW)
    * Event 14: Effectiveness: Task (SUCCEEDED/FAILED, IN/COMPLETE)

The events in the interaction scenarios are then evaluated. The first two breakdowns provide the most specific criteria, and ask the evaluator to consider whether the player understood their goal, and whether the visual representation of the object that they're interacting with communicated its cognitive affordance.

6.4.2 Evaluating Interaction Scenarios

This second level of the player action framework lists the event codes involved in concrete interaction patterns that describe potential usability problems. The patterns are those derived from the earlier heuristic and issue analysis from Section 6.3 (Deriving the Framework).

Each pattern is structured into three sections: Context, Breakdown, and Outcome. Context describes design aspects that on their own may not necessarily constitute a problem, but which contribute to the other sections. A breakdown is where the problem first occurs, typically where the player makes a mistake in the interaction, but also where the design fails in a way that will produce negative outcome for the player. The outcome section primarily includes outcomes defined in the traditional usability terms of Effectiveness, Efficiency, and Satisfaction.

In the player action framework, each component has a scenario interaction pattern listing the events that are related to it. The evaluator examines each of the events listed in the pattern and determines whether they have been violated or not.

To continue the example from earlier, the evaluator documents a usability incident in the area of controls. The interaction patterns in this area include the following events,

- Event 8: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/-tactic used (UN/SUCCESSFULLY)
- Event 14: Effectiveness: Task (SUCCEEDED/FAILED, IN/COMPLETE)

In this example the player fails to correctly execute the action, so the evaluator notes these events as having been violated.

Along with the component and heuristic, the issue scenario provides the main codified part of each usability issue report. The complete report form is shown in Appendix C.4 (Playthrough Evaluation Report Form).

Representing the issues as coded scenarios facilitates more reliable comparison between different evaluation reports. With traditional usability reports analysts have had to make informal, subjective interpretations and comparisons from freeform text descriptions of varying quality. The greater degree of specificity in the scenarios also provides a greater degree of inspectability. This allows disagreements to be explored in fine detail, and improvements to be made to the scenario event patterns in order to clarify and prevent disagreements in future.

These properties are used in the following sections to compute Any-Two values for reliability in how different evaluators identify, analyse, and report usability problems.

6.5 Analysing Playthrough Evaluation

After the footage of the play session has been reviewed and issues reports completed, the data from all of the evaluators can be analysed.

Most studies discussed in Chapter 2 (Literature Review) were limited by not allowing evaluators to conduct the full process of problem detection, and only considered the inter-rater reliability of categorising a pre-determined set of usability issues. In those cases of a fixed number of pre-determined issues, Cohen's Kappa is an appropriate measure of agreement.

However, for a complete evaluation under realistic conditions where each evaluator can detect a different number of usability problems, the Any-Two metric is the appropriate measure of agreement (Barendregt, 2006; Barendregt and Bekker, 2006; Barendregt et al., 2007, 2006; Hertzum and Jacobsen, 2001, 2003), as discussed in detail in Chapter 2 (Literature Review).

Any-Two measures how well pairs of evaluators agree with each other, represented as a percentage averaged across all evaluators, and defined as,

> "...the number of problems two evaluators have in common divided by the number of problems they collectively detect, averaged over all possible pairs of two evaluators."
> Hertzum and Jacobsen (2001)

i.e., the mean of $\frac{|P_i \cap P_j|}{|P_i \cup P_j|}$ over all $\frac{1}{2}n(n-1)$ pairs of evaluators.

Where P_i and P_j are the sets of problems identified by evaluator i and j, and n is the number of evaluators.

All of the analyses in playthrough evaluation use this same metric.

6.5.1 Standardised Grouping Ameliorates the Matcher Effect

In order to compute Any-Two, it is first necessary to group together similar reports from multiple evaluators. The literature on the evaluator effect reported earlier in Chapter 2 (Literature Review) showed that this is another subjective and unreliable stage of evaluation, which is primarily due to a lack of formal procedure to follow. It is usually left for groups of evaluators to discuss in private and negotiate a compromise in order to arrive at a single dominant interpretation for multiple disparate evaluations. Hornbæk and Frøkjær (2008) call this the "matcher effect", similar to the the evaluator effect.

The lack of reliability in this stage of the evaluation process is similar to the other causes of poor reliability throughout heuristic evaluation: the evaluator effect where evaluators and

matchers necessarily must make subjective interpretations based on their different play and evaluation experiences.

Playthrough evaluation follows the definition from Nielsen (1994b) that restricts a problem to be attributable to a single aspect of design. In order to standardise the grouping process in playthrough evaluation problems are grouped together by the task the problem appeared in. This provides a more sound basis for the metrics than if the similarity of problems was performed subjectively.

6.6 Conclusion

This chapter described the theory and process used to derive the playthrough evaluation framework. Heuristics and issues from Chapter 4 (Testing Heuristic Evaluation for Video Games) were decomposed into constituent parts. The parts were then coded in a unified way across both heuristics and issues as discrete interaction events. This provided a common and explicit terminology to construct interaction scenarios as templates for evaluation. This terminology helps evaluators unambiguously specify interaction events in a scenario that are problematic. The coded format enables more precise evaluation, analysis, and improved inter-evaluator reliability. What's more it facilitates critique of the methodology by making explicit a more complete procedure of evaluation.

Chapter 7 (Testing Playthrough Evaluation) describes a series of studies to test the inter-evaluator reliability of playthrough evaluation, and compare it against the reliability of a more traditional heuristic evaluation.

Chapter 7

Testing Playthrough Evaluation

7.1 Introduction

This chapter presents a series of studies to test playthrough evaluation.

The studies consists of two phases,

- In phase 1, playthrough evaluation was tested using pre-recorded reference footage of a play session. Participants all evaluated the same footage.
  Metrics for problem discovery were computed for the playthrough evaluation framework.
  Metrics for problem analysis were computed for the use of playthrough events and heuristics to categorise the issues discovered.
- In phase 2, playthrough evaluation was tested in a more realistic way, where each participant evaluated their own playthrough of two different games.
  Furthermore, each participant evaluated one game using playthrough evaluation and one game using heuristic evaluation.
  Metrics for problem discovery and analysis were computed for both methods, using both playthrough events and heuristics.

22 novice evaluators participated in all of the studies.

The core data explored in this chapter are the standard metrics for problem discovery and analysis when using the playthrough evaluation framework,

- Thoroughness
- Reliability
- Validity
- Effectiveness

The specific research questions asked are,

1. What are the metrics of playthrough evaluation when each participant conducts the evaluation on the same pre-recorded user test session?
2. What are the metrics of playthrough evaluation when each participant conducts the evaluation on their own unique user test session?

3. What are the causes for the differences between these conditions?

Finally the results are considered, with discussion about the reasons for them, and their implications for the value of playthrough evaluation.

7.2 Studies

In order to test the playthrough evaluation framework, a series of usability evaluation studies were conducted,

- Section 7.2.1 (Within Method: Pre-Recorded User Test)
  Initially a within-method study was conducted using playthrough evaluation to evaluate footage of a single pre-recorded user test session of the game *Mirror's Edge*. This tested the validity of the method when used under known conditions, as each evaluator conducted the evaluation on footage of the same playthrough.
- Section 7.2.2 (Between Methods: Playthrough Evaluation and Heuristic Evaluation)
  Following this, further between-methods studies were run using both heuristic evaluation and playthrough evaluation to evaluate 2 typical first-person shooter games, *Aliens Vs. Predator* and *Haze*.

Inter-evaluator reliability was computed for the the evaluations using Any-Two as described in Chapter 6 (The Playthrough Evaluation Framework). Testing the same pre-recorded user test session was expected to produce greater inter-evaluator reliability than testing separate, individual play throughs.

Participants

Participants were recruited by email with the assistance of course leaders from various undergraduate and postgraduate degree courses at the University of Sussex and Brighton University.

All participants were gamers with experience of console first-person shooter, and were students on the following courses:

- BSc (Hons) Computer Science (Games)
- BSc (Hons) Business Computer Systems
- MSc Human-Computer Interaction
- BA (Hons) Digital Media

Topics taught in these courses include:

- Usability Evaluation
- Heuristic Evaluation
- Interface Design
- Game Design
- Interaction Design

- Human-Computer Interaction
- Human Systems
- Ergonomics

24 participants selected, but 2 cancelled, so only data from 22 were used (3 female, 19 male). The mean age was 21.55 years.

Training and evaluation lasted approximately an hour and a half for the first study, and approximately two and a half hours for the second study. Each participant was reimbursed £20 for the total four hours.

7.2.1 Within Method: Pre-Recorded User Test

In the first stage participants used only playthrough evaluation to evaluate pre-recorded footage of a play session.

Participants used the playthrough evaluation framework as a guide to help the discover issues. Once candidates issues were identified participants then used the framework to analyse them, using both heuristics and playthrough events.

This test evaluated the method's ability to discover and analyse issues in a known, reference play session only.

7.2.1.1 Apparatus

Evaluators reviewed footage of a pre-recorded user test session of the author playing *Mirror's Edge*. The session was recorded in a usability lab designed to represent a typical home gaming environment, with the player seated on a large sofa, playing the game on a widescreen television with an Sony PlayStation 3 console.

Footage of the game was captured by routing the console's video output simultaneously to the television and to a video capture card, via a splitter box. Full body video footage of the player was also captured with a digital video camera connected to the same computer recording the game. This provided evaluators with a visual display of the players' body language, facial expressions, and verbal utterances. Both video feeds were composited together using custom software.

Evaluators reviewed the pre-recorded footage on a widescreen computer, using QuickTime software to control playback of the movie file.

7.2.1.2 Procedure

Each evaluator conducted their evaluations independently. A facilitator was available if the player needed help outside the scope of the study.

Training

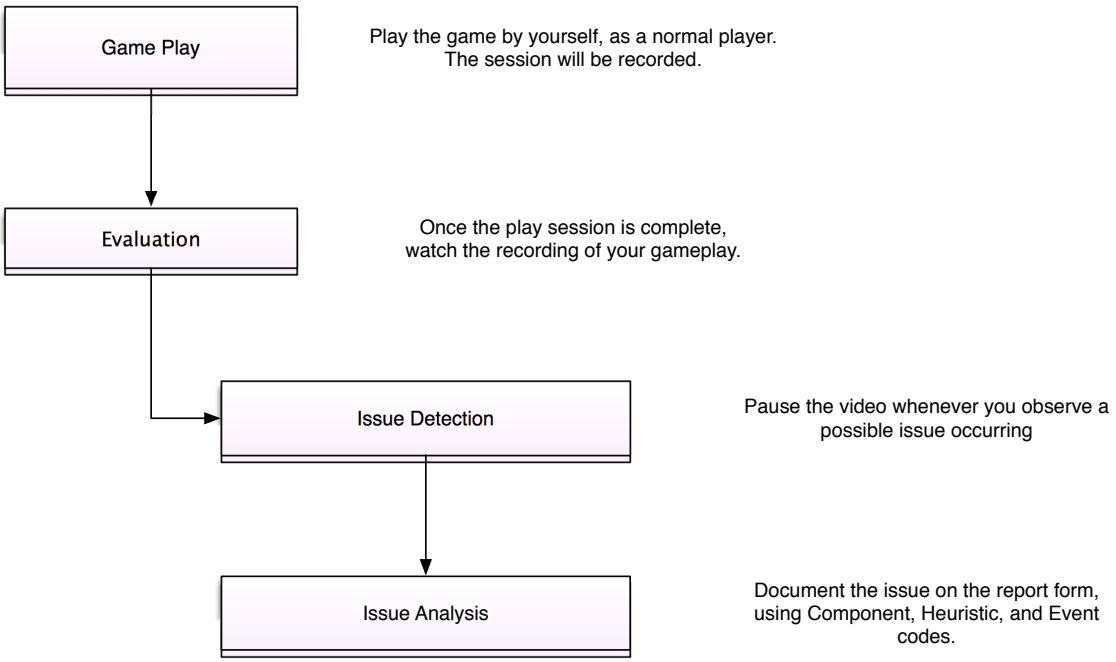
The first stage of the evaluation was to ensure that participants understood and could apply the usability evaluation method correctly.

The purpose and procedure were discussed with the evaluators who were provided with the information in Appendix C.1 (Playthough Evaluation Procedure) which described the method and how to apply it, along with a template form showing an example of a completed report.

The procedure is visually represented by the following diagrams:

- Fig. 7.1 ("Playthrough Procedure - Overview") on this page
- Fig. 7.2 ("Playthrough Procedure - Issue Detection") on the next page
- Fig. 7.3 ("Playthrough Procedure - Issue Analysis") on the following page

Figure 7.1: Playthrough Procedure - Overview



Participants reviewed the documentation, and asked questions for any points that needed clarification. Following their review of the material they were asked to describe the procedure, without referring to the documentation. This gave the moderator an opportunity to check whether they had correctly understood how the method was to be used.

Testing

Following the training, participants watched pre-recorded footage of a user test session of the author playing *Mirror's Edge*, and applied playthrough evaluation by transcribing the events they observed.

7.2.1.3 Analysis

The analysis compared the transcribed data from all of the participants' reports, and was implemented following the procedures described in Chapter 6 (The Playthrough Evaluation Framework).

As the same reference footage of *Mirror's Edge* had been coded by all 22 participants, it was expected that the coding should be substantially the same between all of the evaluators.

Figure 7.2: Playthrough Procedure - Issue Detection

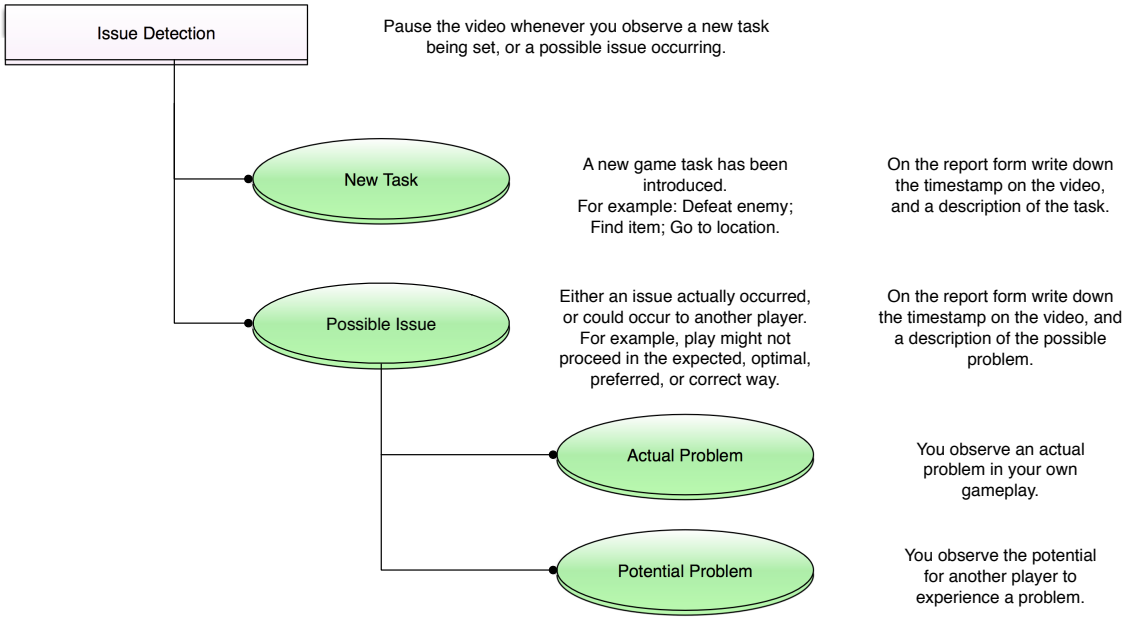
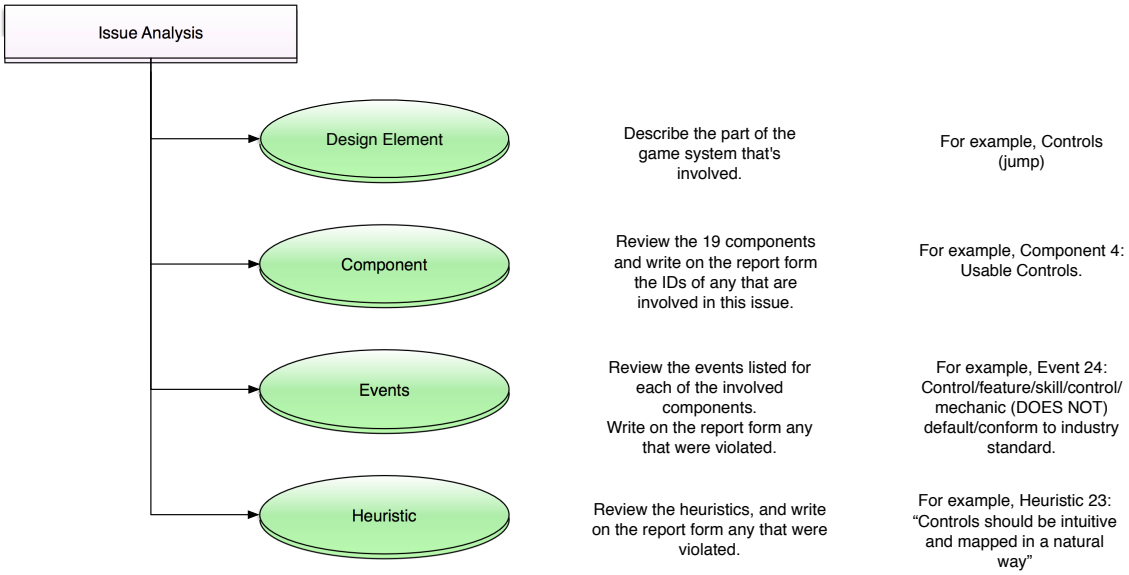


Figure 7.3: Playthrough Procedure - Issue Analysis



All 231 combinations of evaluator pairs were computed with the Any-Two measure.

Playthrough evaluation defines an adaptation to traditional heuristic evaluation, as described in Chapter 6 (The Playthrough Evaluation Framework). During this initial stage of the process, evaluators examined the video footage in a systematic way referring to the list of components and heuristics as a resource for *prospective problem detection* of *potential* issues. The list of components and heuristics acts as a prompt for the evaluator to notice a breakdown in the system design that violates one or more of the heuristics, but which does not necessarily result in an actual negative outcome in the footage being reviewed. For example, the controls did not conform to industry standard, and although this did not cause the current player any particular trouble, the evaluator recognised that this design breakdown could *potentially* cause a problem for another player.

Alternatively, evaluators may notice an *actual* problem occurring, typically by first observing a negative outcome that they had experienced. In this case evaluators reviewed the components and their heuristics in the player action framework, and recorded any that were involved in the candidate incident.

This stage of the process is similar to a traditional prospective heuristic evaluation, but with the additional opportunity for detailed inspection provided by the post-gameplay video review. As all of the evaluators observed the same footage and applied the same process with the same heuristics, inter-evaluator reliability was computed for their use of heuristics to describe the same problems.

In addition to the post-gameplay video review, playthrough evaluation also provides a lower level of structured analysis. Once a candidate problem had been detected, the player action framework was then used for *retrospective* analysis. In this stage evaluators described the interaction scenario using detailed, atomic events associated to each of the higher level heuristics identified in the previous stage. Each heuristic provided a list of events that the evaluators considered for violations, in a manner similar to the way the heuristics were considered for violations. All event violations were documented by the evaluators for each of the heuristics they had identified. Once again, as all of the evaluators had observed the same footage, and applied the same method with the same heuristics and sets of events, inter-evaluator reliability was computed for their use of the novel events to describe the same issues.

The following section shows the separate Any-Two values for the playthrough events and heuristics.

7.2.1.4 Results

Problem Discovery

Data for problem discovery are shown in Table 7.1 ("*Mirror's Edge* problem discovery") on the next page.

Of particular note is the relatively low value for Thoroughness, but high value for Validity. The Validity value shows that almost all issues were detected by more than one evaluator, suggesting that the problems were likely to affect other players as they were not isolated incidents discovered by a single evaluator. However, the low Thoroughness shows that in most cases evaluators only detected a small number of the total problems that players could encounter.

Table 7.1: *Mirror's Edge* problem discovery

Game	Metric	Any-Two
Mirror's Edge		
	Thoroughness	0.19
	Validity	0.95
	Effectiveness	0.27
	Reliability	24.59

Table 7.2: *Mirror's Edge* problem analysis

Method	Any-Two
Playthrough heuristics reliability	10.45
Playthrough events reliability	18.72

This suggests that many evaluations would be needed to detect most problems in a complex game like the ones used in these studies.

Problem Analysis

Data for problem analysis are shown in Table 7.2 ("*Mirror's Edge* problem analysis") on this page. This shows the reliability figures produced for both forms of problem analysis in playthrough evaluation, firstly for the original heuristics and secondly for the more specific event patterns. The table shows that playthrough evaluation produced low to moderate levels of agreement for problem analysis. This means that even when independent evaluators discover the same issue they classify it using different codes, whether they used playthrough events or heuristics. They observed the same data and identified the same problematic aspect of the game, but these results suggest that they came to different conclusions about exactly what had happened.

What's more, in all cases the inter-evaluator reliability of problem analysis in playthrough evaluation using events is substantially greater than that when using heuristics. This result is encouraging as there are more events than heuristics, and so we would expect worse reliability if the coding was performed by chance alone. The data suggests that the scenarios defined by the player action framework helps evaluators reach similar conclusions to one another, despite having a wider range of events to choose from than heuristics.

7.2.1.5 Discussion

The results show that participants independently evaluate the same footage in similarly ways to each other. However, the evaluator effect is still evident, especially in the low values of problem discovery thoroughness and problem analysis reliability. There were at least two factors that influenced these results: participant's evaluation and gameplay expertise.

Participants' *Evaluation* Ability Affects Their Experience

The evaluators used in this study had prior experience of design and evaluation, but were not professionals, and should be considered as *novice* evaluators. They did not have the expertise to be able to understand the player experience and make good predictions about what problems other players would experience.

While they could not change their own ability level, an expert evaluator should be able to make stronger predictions about players with different ability levels, based on observation of a wide range of different players.

The original intention for the playthrough evaluation framework was to provide more domain-specific and structured support for novice evaluators. In this respect the framework achieves its purpose. In contrast, heuristic evaluation was designed as an *expert* evaluation method. The problems with reliability presented in Chapter 2 (Literature Review) highlight the fact that the evaluator effect is more pronounced for using novice evaluators.

Participants' *Play* Ability Affects Their Experience

Every player has their own tendencies and proficiencies in playing games like these. In an first-person shooter game the evaluator does not have the luxury to take time to slowly, carefully, and systematically explore the entire system in the way they might for a traditional domain. They must react quickly and efficiently. As a consequence they will use their best abilities to play as optimally as possible. Optimal play is not the best way to fully understand and explore a game from the perspective of evaluation, however.

Furthermore, games should be designed to accommodate a variety of different play styles and player ability levels. Indeed, these design principles can be found in some of the original reference heuristics in the literature. For a single evaluator to conduct a very thorough evaluation they would need to experience the game from the varied perspectives of these diverse styles and abilities. A situation that might seem reasonable for an experienced player might seem unreasonable for a beginner. The evaluators in the studies all had gaming experience before, but their ability levels were heterogeneous. While some could be considered *hard core* or *expert players*, others were *novice players*. When considering the evaluator effect for video games, the evaluator's gaming expertise may be an additional aspect to take into account.

Limitations

This study only tested the evaluators' ability to evaluate a pre-recorded video of gameplay. This makes the process of evaluation considerably simpler, and the results shown here do not necessarily suggest that a real evaluation would perform as well. Nonetheless, the results were encouraging in that for this initial condition they showed that evaluators coded issues more reliably when using the novel events from playthrough evaluation than when just using heuristics alone. This provides some support for pursuing further studies exploring playthrough evaluation under more realistic evaluation conditions.

In the following studies participants played and evaluated the games independently of one another. In that situation although each participant evaluated the same game, their individual play through was unique. There was no guarantee that each evaluator would review the same

phenomena in the game, and so the reliability of their evaluations was expected to be lower than seen with the *Mirror's Edge* experiment.

7.2.2 Between Methods: Playthrough Evaluation and Heuristic Evaluation

7.2.2.1 Design

Evaluations were performed on two different games where, unlike in the previous study, participants played and conducted the evaluations independently of one another. Furthermore the evaluations were performed with two methods:

1. Playthrough evaluation using events novel to the method.
2. Traditional heuristic evaluation using heuristics alone.

Inter-evaluator reliability for each separate method was computed, and comparisons drawn between the two.

7.2.2.2 Participants

The 22 participants from the previous study also conducted the evaluations in this study. The participant population was randomly divided into two groups. All participants used both methods, but the presentation of the methods to the groups was counterbalanced to prevent order effects. i.e., the first group evaluated with heuristic evaluation first, then with playthrough evaluation; the second group evaluated with playthrough evaluation first, then with heuristic evaluation.

Two representative first-person shooter games were evaluated, *Haze* and *Aliens Vs. Predator*. All participants evaluated both games, but the presentation of the games was also counterbalanced. i.e., the first group evaluated *Haze* first, then *Aliens Vs. Predator*; the second group evaluated *Aliens Vs. Predator* first, then *Haze*.

This combination of methods and games gave four groups of evaluation ordering,

1. The first set of evaluators initially tested *Aliens Vs. Predator* with heuristic evaluation.

   The first set of evaluators then tested *Haze* with playthrough evaluation.

2. The second set of evaluators initially tested *Haze* with heuristic evaluation.

   The second set of evaluators then tested *Aliens Vs. Predator* with playthrough evaluation.

3. The third set of evaluators initially tested *Aliens Vs. Predator* with playthrough evaluation.

   The third set of evaluators then tested *Haze* with heuristic evaluation.

4. The fourth set of evaluators initially tested *Haze* with playthrough evaluation.

   The fourth set of evaluators then tested *Aliens Vs. Predator* with heuristic evaluation.

7.2.2.3 Apparatus

The games were tested on a Sony PlayStation 3 (*Haze*) and Microsoft Xbox 360 (*Aliens Vs. Predator*). Evaluators worked in a comfortable environment designed to simulate a living room, seated on a large sofa and played the game on a widescreen television. Footage was captured in the same way as for the previous *Mirror's Edge* study. Similarly, during the playthrough evaluation, participants reviewed the video footage of their playthrough in the same manner as with the previous study.

7.2.2.4 Procedure

Each evaluator conducted their evaluations independently. A facilitator was available if the player needed help outside the scope of the study. While playing the games, participants were asked to think aloud, and were recorded on a video camera. Each evaluator applied both methods, described separately in the following sections.

Playthrough Evaluation

Participants were given an overview of the method, shown in Appendix C.1 (Playthough Evaluation Procedure) which explained the method and terminology being used. The procedure was discussed with the participants to verify that they understood their meanings and how they would use the method.

Participants played their initial test game (either *Haze* or *Aliens Vs. Predator* depending on the group the participant was assigned to). The audio and video output of the game was recorded simultaneously with a video recording of the player. The play session lasted approximately 20 minutes, and covered the first level of the game. After the play session was finished, participants were shown the video footage of the game, and told to apply playthrough evaluation by transcribing the usability events using the player action framework.

The participants had previously been taught to use playthrough evaluation in the previous study on *Mirror's Edge*, so only a brief reminder of the method was given. The evaluation procedure followed the same as previously described in the earlier section.

The evaluators were free to skip through the video as they liked, to pause, rewind, etc. Applying the method took approximately 60 minutes.

Heuristic Evaluation

As with playthrough evaluation, the first stage of the heuristic evaluation was a brief review of the method to orient the participants. The evaluators were provided with the information described in Appendix C.2 (Heuristic Evaluation Procedure) which explained the method and specific heuristics being used. Each heuristic was discussed with the participants to verify that they understood their meaning and how they would be used in the method.

Following initial training, evaluators played the main test game (either *Haze* or *Aliens Vs. Predator* depending on the group the participant was assigned to). When an issue was encountered the evaluator was free to pause the game and document it at a convenient point. Sometimes, particularly during periods of intense action, this was inconvenient and pausing

the game would have interrupted their sense of immersion. This is especially important for things like spatial and situational awareness; pausing the game during a firefight would disrupt the player's embodied awareness of the environment, and make the return to the game jarring as they have to reorient themselves to the space and its contents. If necessary in these cases the evaluator paused the game after the fight.

7.2.2.5 Analysis

Once all of the participants had completed evaluations of their own game sessions the data were analysed for inter-evaluator reliability. Each of the evaluation methods was analysed separately.

In the previous study with *Mirror's Edge* where the same reference footage had been coded by all 22 participants, reliability was expected to be substantially the same between all of the evaluators. Differences were expected to be more substantial when conducting individual evaluations on *Aliens Vs. Predator* and *Haze*, however, as each evaluator experienced the game in their own unique play through.

Participants played and reviewed *Aliens Vs. Predator* and *Haze* using both heuristics and events in the playthrough evaluation condition, but only used heuristics in the traditional heuristic evaluation condition.

7.2.2.6 Results

Metrics are reported separated for problem discovery and problem analysis.

Problem Discovery Was Valid, but Not Thorough

Metrics for the two games are shown in:

- Table 7.3 ("Problem discovery with playthrough evaluation framework ") on the next page
- Table 7.4 ("Problem discovery with heuristic evaluation ") on the following page

In both games validity was very high, especially in the playthrough evaluation condition. This metric indicates how many issues detected by the method were actually experienced by other players. In almost all cases when evaluators reported issues they were also encountered by other players in the study too.

Despite the very high validity, thoroughness was low at just 0.29, and 0.39, for *Aliens Vs. Predator* and *Haze* respectively.

This means that evaluators tended to discover different issues to one another. One possible reason for the low values could be due to the evaluator effect. Although the method defined clear guidelines for systematic evaluation many participants conducted the evaluation in a less formal manner. The facilitator of the study observed that participants often did not pause the video when problems were evident in the footage, which somewhat undermines the procedures defined. Nonetheless, even if they had taken more care to follow the process it may not have produced significantly different results. Evaluators seemed to be performing a kind of filtering

Table 7.3: Problem discovery with playthrough evaluation framework

Game	Metric	Any-Two
Aliens Vs. Predator		
	Thoroughness	0.29
	Validity	0.92
	Effectiveness	0.27
	Reliability	15.53
Haze		
	Thoroughness	0.39
	Validity	0.96
	Effectiveness	0.37
	Reliability	33.58

Table 7.4: Problem discovery with heuristic evaluation

Game	Metric	Any-Two
Aliens Vs. Predator		
	Thoroughness	0.16
	Validity	0.80
	Effectiveness	0.12
	Reliability	11.15
Haze		
	Thoroughness	0.27
	Validity	0.76
	Effectiveness	0.21
	Reliability	32.08

process while reviewing the footage, where that they didn't see the need to pause and review the session unless a significant problem occurred.

This meant that, for example, a skill was used unsuccessfully and resulted in a task failure, but the participant did not pause the video and document the issue. This is understandable to a certain extent as the evaluator did not experience these kinds of case as "problems" as far as the player experience goes. Breakdowns like this were expected by the players, who did not consider them worth documenting. This relates to an important distinction between the meaning of problem in a usability context and a playability context. Despite the study attending to usability only, evaluators treated the issues from a playability point of view.

This introduces more scope for the evaluator effect as different evaluators have quite different attitudes towards what constitutes a good player experience.

In future studies it may be necessary to emphasise the purpose of the procedure even more so that evaluators focus on usability. Alternative detailed training sessions could be used

to test the participant's ability to thoroughly detect and analyse issues. A useful approach would be to use a benchmark with known issues, like the *Mirror's Edge* footage, and test how well the evaluators are able to use playthrough evaluation to identify and analyse them.

Problem Analysis

During the problem discovery stage, reliability represented the number of problems that evaluators discovered in common with one another. In the problem analysis stage, reliability represented the number of events or heuristics that evaluators used in common to categorise an issue in the same part of the game.

In cases where evaluators discovered problems in the same task and the same design element, reliability was again computed but using a modified form of Any-Two. In this form the metric was calculated based on how many specific event codes the evaluators had in common when they described the issue. For example, one evaluator may have evaluated the following two events to have been violated:

- Event 1: Skill/control/action/mechanic/feature/tactic (IS/NOT) introduced
- Event 3: Player (IS/NOT) provided with opportunity to practice skill/control/action/mechanic/feature/tactic

And if another evaluator had evaluated violations to have occurred in these two,

- Event 3: Player (IS/NOT) provided with opportunity to practice skill/control/action/mechanic/feature/tactic
- Event 8: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/-tactic used (UN/SUCCESSFULLY)

then the intersection computed by Any-Two would be 1 as both evaluators have event number 3 in common. Similarly the union of their events would be 3 as there are 3 different events involved. Any-Two for this case would then be computed as a value of 0.3.

This computation shows how similarly independent evaluators rated issues that they had discovered in common.

Results are shown in the following section.

Reliable Problem Analysis With Playthrough Events

Inter-rater reliability for problem analysis is presented in the following tables, showing the differences in reliability for *Aliens Vs. Predator* and *Haze*,

- Table 7.5 ("Playthrough evaluation problem analysis reliability") on the next page
- Table 7.6 ("Heuristic evaluation problem analysis reliability") on the following page

The data shows that describing issues with the event coding of playthrough evaluation produced relatively good inter-evaluator reliability. Despite the greater number, variety, and specificity of event codes, evaluators were still able to produce reliable results when conducting the analysis with playthrough events.

Table 7.5: Playthrough evaluation problem analysis reliability

	Any-Two
Game	
Aliens Vs. Predator	17.02
Haze	15.05

Table 7.6: Heuristic evaluation problem analysis reliability

	Any-Two
Game	
Aliens Vs. Predator	7.53
Haze	4.08

7.3 Discussion

The nature of the Any-Two calculation, and the standardised grouping process described in Section 6.5.1 (Standardised Grouping Ameliorates the Matcher Effect) facilitates analysis at a level of detail much greater than provided by heuristic evaluation. In traditional heuristic evaluations, where disagreements are resolved privately by informal discussion, any potential for improving the method is lost as the source and resolution to these disagreements are not reported in the literature.

The novice evaluators in these studies strongly based their evaluations on their own playthrough, and had difficulty predicting what the experience would be like for other players. There were several occasions where, for example, one evaluator might comment that a particular task or interface was problematic and likely to cause a lot of trouble for other players, while another evaluator considering precisely the same design aspect felt that it was well made and would not cause any difficulties.

These disagreements are to a certain extent to be expected, particularly when using novice evaluators. This is the fundamental limitation of the evaluator effect. Novice evaluators in particular lack the necessary resources to be able to predict other users' problems. This expertise principally comes through experience of observing numerous user test sessions.

Despite this, even novice evaluators were able to use playthrough evaluation to produce more reliable results than when using heuristic evaluation. This is promising for future work, which could explore the potential for expert evaluators to use the method too. The current studies have already demonstrated the potential for this method by improving performance at the novice level of expertise. This holds promise for the research community by testing longitudinal studies in the future, where novices can progressively be trained to become playthrough evaluation experts. It will be interesting to see how their performance improves once they have

developed expertise at prediction, based on observation of many other user test sessions.

Hierarchic Problem Matching Might Improve Results

In future iterations of the playthrough evaluation framework it may be more valuable to group problems in a hierarchical fashion. As discussed earlier with regard to heuristics in Chapter 4 (Testing Heuristic Evaluation for Video Games), forcing evaluators to use a single code to describe an issue may not be a reasonable approach.

The structure of the player action framework, like the user action framework, is intended to allow evaluators to reach the same terminal nodes via a number of different paths. This effectively bypasses the problems that a flat, linear and singular structure present, by allowing evaluators to reach a final node via a series of hierarchical levels of increasing specificity. Similarly it might be better to recognise that when it comes to similarity of problems, forcing them into exclusive combinations is an unnatural and unrealistic approach. A more natural understanding of similarity would encompass multiple dimensions, acknowledging that there are cases where contexts involve multiple tasks, some of which may be more or less closely related.

7.3.1 Limitations

Section 2.5 (Metrics) discussed the ways that a usability evaluation method's performance can be measured. The most common metrics for problem discovery and analysis were used in the studies in this thesis:

- Thoroughness
- Validity
- Effectiveness
- Reliability

However, an important measurement was not taken into account that is important to consider when discussing discovery and analysis resources: time.

Playthrough Evaluation Requires More Time

The first study lasted approximately one and a half hours for training and evaluation of the pre-recorded footage. The second study lasted approximately two and a half hours. In this time the participants played and evaluated two games, one with playthrough evaluation and one with heuristic evaluation. These studies did not explore Actual Efficiency using equation (eq. 8) from E. L.-C. Law and Hvannberg (2004b). However, the total time to play and evaluate each game using playthrough evaluation was informally noted as taking approximately 2-3 times as long as for heuristic evaluation.

Commercial Evaluations May Be Time Critical

Commercial usability evaluations in the real world are usually constrained by the available resource budget. Time is often a critical resource, especially in formative cases where rapid iteration is desirable. As such, although playthrough evaluation performed well in regard to the other standard metrics, the results provided may require a prohibitively expensive investment in time for most industry practitioners.

As a case in point, during commercial work with Vertical Slice the evaluation team typically produced a rapid "hot feedback" email at the end of each day's testing. This was a brief and cursory summary of the main issues observed, focussing on the top priority, most severe, and frequent issues. Providing this feedback gave the developers the opportunity to immediately respond and move forward in their development process without having to wait for a formal report or presentation to be produced. In some cases where the issues had a serious effect but were simple to resolve, the production team were able to fix them overnight and deliver a new build for testing on the following day.

More Time May Be Reasonable for Summative Evaluations

Nonetheless, the focus of this thesis and the playthrough evaluation framework was on summative evaluations. These are normally conducted on products that have already been finalised and are not subject to rapid iteration as in the formative stages of the lifecycle. In these cases it may be feasible to invest a greater amount of evaluation resources, such as using more time or involving a larger number of evaluators to test inter-rater reliability.

This subject is considered further in Chapter 8 (Conclusions), where contributions are discussed for practitioners and researchers.

7.4 Conclusions

This chapter looked to explore the standard metrics for the performance of playthrough evaluation. The following conclusions are made, for problem discovery with playthrough evaluation,

- Mean effectiveness was twice as high overall when evaluators used playthrough events than playthrough heuristics.
- Thoroughness values were low throughout all of the studies.
- Validity was very high throughout.
- Reliability was modest for both conditions, but consistently higher for playthrough events.

To summarise, in playthrough evaluation using events produced better results than using heuristics for first-person shooter video game usability evaluation. Furthermore, the level of detailed analysis that the method gave provides much more insight into the specific problems encountered. This degree of analysis can facilitate a critique of the method at a very detailed level, with specific recommendations for improving the procedures and reliability of results they generate. Despite these benefits, the evaluator effect was still clearly evident, especially in regard to the novice evaluators poor ability to predict the problems that other players would

encounter. Although the framework provided structured guidance and additional resources to assist the evaluation, these were insufficient to compensate for the individual differences and lack of expertise in the novice evaluators.

The following chapter concludes this thesis by reviewing the original research questions, reflecting on how well playthrough evaluation achieved its aims, pointing out limitations, and highlighting potential future work.

Chapter 8

Conclusions

This chapter reviews the research questions asked, and considers how well they have answered in the thesis. A summary of novel contributions is presented, and a case made for their value to the research community. Limitations are noted, and research avenues for further work are suggested.

8.1 Research Questions

The following research questions were posed by this thesis in Chapter 1 (Introduction),

1. How reliable is heuristic evaluation for first-person shooter games?
   (Chapter 4, Testing Heuristic Evaluation for Video Games)
2. What are the causes of reliability problems?
   (Chapter 5, Exploring Evaluation Resource Specificity)
3. How can a novel framework be derived to address these problems?
   (Chapter 6, The Playthrough Evaluation Framework)
4. How well can a novel usability evaluation method within this framework improve on the reliability of heuristic evaluation?
   (Chapter 7, Testing Playthrough Evaluation)

Chapter 2 (Literature Review) showed the deficiencies in reliability for user test in general and heuristic evaluation specifically, and the latter studies developed a novel approach which demonstrated an improvement in evaluation reliability. Each of the individual chapters in summarised in the following section.

8.2 Summary of Chapters

Chapter 2 (Literature Review)

The literature review considered how usability and evaluation has been addressed for traditional domains as well as video games. A particular emphasis was placed on discussing how usability problems are discovered and analysed. The evaluator effect was identified as a key concern, describing the differences in evaluation results produced by independent evaluators. Heuristic

evaluation was introduced as the most widely employed method for evaluating video games, though concerns were made about the lack of formality in the processes used.

Two novel terms were defined to describe the different implicit ways that problem discovery and analysis is conducted in heuristic evaluation: Feed-forward (prospective); and Post-hoc (retrospective). Prospective evaluation is heuristic-focussed, where the heuristics help to guide the evaluator to *discover* problems. Retrospective evaluation only uses the heuristics as a form of *analysis* to describe the issues that are typically discovered in a more free-form way.

Furthermore, by standardising evaluation procedures it was hypothesised that the evaluator effect could be ameliorated. Some degree of difference in evaluation will nonetheless remain, due to the inherent subjectivity involved in interpreting evaluation data, and the inevitable differences in evaluator expertise. However, by employing strategies to identify and control the effect levels of inter-evaluator reliability should improve.

Chapter 4 (Testing Heuristic Evaluation for Video Games)

The first empirical study in this thesis began by collecting user test data from a single player first-person shooter game, *Aliens Vs. Predator*. During the initial testing it was noted that observers made different reports to one another, so further analysis was proposed to explore the differences. Following the procedure used by Nielsen (Nielsen, 1994a) a heuristic evaluation was conducted using 146 heuristics from the literature, and rating them against 88 issue reports recorded during user testing. Three evaluators collectively made 38,544 ratings of the issue–heuristic paris, and Krippendorff's Alpha was computed at a value of 0.343, which indicated systematically poor inter-rater reliability. Continuing with Nielsen's approach, the data were explored further with principal components analysis. The results suggested that although evaluators disagreed on the ratings to assign to specific heuristics, there were a core of 19 areas that they tended to all address.

Chapter 5 (Exploring Evaluation Resource Specificity)

Although the previous chapter had explored the results of the evaluation, it was still unclear why the differences had occurred. Evaluators were interviewed and discussed how they had conducted the evaluation, which revealed that they were often focussing on different aspects of the heuristics and issues to one another. The differences in ratings was suggested to be due to a combination of the ambiguous phrasing of the individual heuristics, complexity of the user tasks involved, and the lack of structure to help guide the evaluators in relating the heuristic criteria to the tasks.

By considering the content and presentation of the heuristics themselves, a novel insight was made that the homogenous term "heuristic" is used for three different forms, termed: *design principles*, *abstract reflection*, and *outcome analysis*. Design principles are phrased as *positive guidelines*, which would be particularly useful during formative evaluation to aim the development to include positive design aspects. Abstract reflective heuristics have more use as a way to remind an expert evaluator about the general themes to consider, but offer very little in the way of resources for specific problem *discovery* or *analysis*. Analytical heuristics refer to *negative outcomes*, and offer utility particularly during summative evaluation through

their use of implicit criteria that can indicate violation. However, they can also be problematic when multiple criteria are used in the same heuristic, as evaluators are given no support in how to decide which criteria would constitute a violation of the heuristic as a whole.

Content analysis was introduced as a way of unpacking the criteria in heuristics as a way to make explicit the design and evaluation knowledge contained in them.

Chapter 6 (The Playthrough Evaluation Framework)

Content analysis was systematically applied to the heuristics and issues from the earlier studies to derive a common set of terms for describing user test issues and the specific criteria needed to evaluate them. This novel coding scheme was used to compose interaction scenarios for each of the principal components involved in evaluating first-person shooter games. These represent a template of expected interaction between the player and the game, and can be used during evaluation to explicitly indicate the appropriate criteria to evaluate for each incident.

The playthrough evaluation methodology was described, including the procedure for applying the method and analysing the results. Participants begin the process by playing the game in a natural, non-evaluators way, and the footage of their play session is recorded for subsequent analysis. This provides them with a first-hand experience of the gaming experience prior to the evaluation proper. Following the play session, the participant evaluators review the video and systematically apply the playthrough evaluation method to each new task as it occurred in the game footage. When a candidate problem is observed evaluators select one or more of the principal components derived from Chapter 5 (Exploring Evaluation Resource Specificity), and consider the interaction scenarios they contain. Each event in the scenario is evaluated in terms of whether it has been violated or conformed to. After all candidate issues have been evaluated in this way, inter-evaluator reliability metrics are computed using the Any-Two measure of agreement.

Chapter 7 (Testing Playthrough Evaluation)

In order to test the method, a study was run to measure playthrough evaluation in terms of the standard metrics. 22 novice participants evaluated three games, *Aliens Vs. Predator*, *Haze*, and *Mirror's Edge* using a *prospective* heuristic evaluation, as well as playthrough evaluation with events and heuristics.

Metrics were computed separately for problem discovery and problem analysis.

For problem discovery the mean thoroughness values were computed as 0.19, 0.29, and 0.39 for *Mirror's Edge*, *Aliens Vs. Predator*, and *Haze* respectively. These relatively low values of thoroughness indicate that any given evaluator only detected a small proportion of the total number of issues in the system. It is of interest that the values computed for *Mirror's Edge* were the lowest, as this was the first of the three games to be tested, and the evaluation was only conducted on pre-recorded footage. For the other two games the evaluators first *played through* the game by themselves. As *Mirror's Edge* was the first game to be tested, this lower value may be due to the evaluators' inexperience with the method. Presentation of the other two games was counter-balanced to prevent order effects, so the differences in values cannot be due to the evaluators' experience.

Problem detection reliability was computed as 24.59%, 15.53%, and 33.58%, for *Mirror's Edge*, *Aliens Vs. Predator*, and *Haze* respectively. A similar pattern can be seen with thoroughness, where *Aliens Vs. Predator* produced the lowest values, and *Haze* produced the highest.

Reliability was also computed for problem analysis. In cases where multiple evaluators discovered an issue in the same design element the average Any-Two reliability when using playthrough events was 18.72%, 17.02%, and 15.05% for *Mirror's Edge*, *Aliens Vs. Predator*, and *Haze* respectively.

The literature on game evaluation has not previously report these metrics, so direct comparisons cannot be made to other methods. Nonetheless, as presented in Chapter 2 (Literature Review), data are available for evaluation studies in traditional domains. In particular, Hertzum and Jacobsen (2001) computed Any-Two values for four studies that used novice evaluators:

- Hertzum and Jacobsen (1999) reported a cognitive walkthrough of a Web-based library system conducted by 11 CS graduate students, with an Any-Two value of 17%.
- Nielsen and Molich (1990) conducted a heuristic evaluation on a savings application, using 34 CS students, with an Any-Two value of 26%
- Connell and Hammond (1999) conducted two heuristic evaluations. The first study of a hypermedia browser involved 8 undergraduates and produced an Any-Two value of 9%
- The second examined an interactive teaching application by 8 psychology undergraduates, and produced an Any-Two of 8%.

The values produced in the playthrough evaluation studies are within the range seen in other domains.

The intention for the playthrough evaluation framework was to create a novel methodology for games that could ameliorate the evaluator effect. The results from Chapter 7 (Testing Playthrough Evaluation) were reasonable and comparable to those seen in other usability evaluation methods, especially discount methods such as heuristic evaluation. Furthermore the structure of the method allows evaluations to describe interactions in detail, using events that are specific to video games.

8.3 Limitations

The Evaluator Effect Needs to Be Better Understood

The data still showed relatively low values, especially for Thoroughness. The main factors to influence these values were evaluator and *player* experience. In addition to the evaluator effect, when conducting an evaluation of a video game, the *evaluator's ability as a player* is an important factor that determines the results of the evaluation. Analysis of the data and process suggest that this may be due to individual differences, and subjective bias when using one's own experience of the game as a benchmark against which to make predictions. Evaluators seemed to have difficulty imagining how other players could experience the game in a way different to their own. This is a highly significant aspect of the evaluator effect that should be studied in detail in future work. As a research community we need to understand what factors influence evaluators' decisions, and how to train evaluators to make better decisions.

The playthrough evaluation framework attempted to provide more structured discovery and analysis resources as a way to ameliorate the effect. Chapter 7 (Testing Playthrough Evaluation) showed that the method is still affected, but these resources make the evaluation more detailed.

8.3.1 Limited Scope

This thesis has only considered a limited number of criteria, derived from a relatively small number of heuristics. The studies involved a reasonable number of participants, though only a small number of video games, and all from a single genre.

Heuristic evaluation as a method in abstract is general, but the specific heuristics used in for any particular domain, platform, or evaluation may be more or less appropriate. So too, playthrough evaluation can be applied to a variety of domains, platforms and specific evaluation projects. As it stands, the event codes included have been demonstrated to be suitable for first-person shooter games, and it is reasonable to expect them to be viable throughout this genre.

In order to constitute a methodological contribution it was necessary to demonstrate the methodology being used in a large number of different cases. The initial database for Chapter 4 (Testing Heuristic Evaluation for Video Games) consisted of 38,544 ratings, using 3 evaluators, 88 issues, and 146 heuristics. This formed the main corpus which the following studies explored in more detail. The 88 issues were all reported from a single game, and so further studies will be able to make knowledge based contributions to add further nuances to the evaluation criteria already described. Additional example transcriptions are expected to shed light on further edge cases, which will strengthen the current knowledge contribution made by this thesis. Furthermore, this mechanism is a strength of the playthrough evaluation framework, whereby additional studies can be compared against the current standard, and improvements discussed and shared with the research community. The shared terminology, exposed procedures and data all ensure that critical appraisal and testing can allow the knowledge contribution of the methodology to improve.

Playthrough Evaluation May Not Be Appropriate for Practitioners

Følstad et al. (2012) reported a survey of usability practice in the real world. Their data suggest a gulf between the research being developed, particularly in academia, and the needs of practitioners in industry.

> "Usability research on forms, formats, and tools does not seem to have much direct impact on analysis practice ...tools developed by the research community tend to be complex (as is arguably the case for SUPEX and UAF) and difficult to learn. Complexity is not compatible with the time demands of the practical evaluation context"

The intention for the playthrough evaluation framework was to address some of the weaknesses of light-weight, discount usability evaluation methods. In particular, the framework was designed with novice evaluator in mind who may require additional assistance and guidance

when conducting an evaluation. The main approaches taken were to increase the specificity of evaluation by adding additional structure and resources for evaluators to use. The results reported in Chapter 7 (Testing Playthrough Evaluation) are encouraging in that they demonstrate the use of the method for novice evaluation of complex first-person shooter games.

However, this thesis did not address the context and requirements of practitioners in the real world. Følstad et al. (2012) point out that,

> "For new analysis support to be successful, it will have to fit the fast-paced analysis context"

Playthrough evaluation does not meet this criterion as it requires additional training and even more time to conduct evaluations than other traditional light-weight methods.

Furthermore, practitioners in the real world are likely to be professionals with a great deal more experience conducting evaluations than the novices used in this thesis. Further work would be need to explore whether different results are seen if expert evaluators use the playthrough evaluation framework.

8.4 Further Work

Woolrych et al., 2011 argue that attempting to compare usability evaluation methods on their own can be potentially misleading. They particularly critique lab-based experimental conditions that do not reflect the complexities of usability work conducted by practitioners in the field. Instead they draw a metaphor of usability work as cooking a meal, where resources are like ingredients, and usability evaluation approaches are like recipes for dishes. Their research agenda is to identify constituent resources or aspects of methods that can be adapted and combined for usability work. Følstad et al. (2012) also note that practitioners often adapt and appropriate a mixture of methods and tools.

The lab-based experimental approach presented in Chapter 7 (Testing Playthrough Evaluation) may be too resource intensive and unrealistic for practitioners to use, especially as a single monolithic method. However, the playthrough evaluation framework is composed of playthrough evaluation and the player action framework, and future work could explore whether individual *ingredients* of the playthrough evaluation framework would be suitable for use by practitioners.

Candidate resources in the playthrough evaluation framework include:

- Video review
- Event codes
- Event scenarios
- Principal components

Post-gameplay video review was an important resource that gave evaluators the possibility to reflect on the footage and provided another opportunity to focus on problem detection. However, the cost of this resource was a large increase in the amount of time required.

Event codes and scenarios are a unique contribution of the framework, with application for both problem detection and analysis. They may additionally have value for task-scenario development and as a way to assist traditional light-weight evaluation. The high degree of specificity for the domain could be particularly useful for criteria-based analysis of issues identified with other, faster methods.

Lastly, the principle components and their associated heuristics were derived specifically for first-person shooter games. This thesis contributes to the established practice of customising heuristics for individual domains, and the components identified suggest a core set of areas appropriate for heuristic evaluation of this style of game. This aspect of the framework is likely to be the most easily adapted for practitioners in the real world, and the most efficient contribution for them to make use of.

8.4.1 Playthrough Evaluation Throughout the Lifecycle

The emphasis of this thesis has been on summative evaluation, and a primarily hygiene-oriented understanding of usability, i.e., the discovery and analysis of usability problems.

In an iterative evaluate-redesign cycle, evaluation needs to be reliable before any new designs should be attempted. Design decisions made on the basis of poor evaluation may fail to address the real problems in the system, and may in fact create further problems for future versions.

However, the playthrough evaluation framework does shows potential to address evaluation during the formative stages of a product. This aspect of evaluation has not been tested in this thesis, though there is no theoretical reason that it should not be viable. In order to explore this potential, playthrough evaluation should be used during pre-production, prototyping and iterative development of a real product in a real-world case study. Such a longitudinal study would be ideally suited to a further Ph.D. programme, post-doctoral fellowship, or other long-term research project. Research questions would be in line with the discussion of usability by Cockton (2012) (considered in detail in Chapter 2 (Literature Review)) and oriented around the following topics:

- How the method is actually used, adopted or appropriated by a real design team.
- What contribution it makes to the overall development process and final product.
- How evaluation can help to create positive value as well as reduce the negative hygiene components.
- How the designed solutions are experienced by players in the real world instead of a lab.

However, to understand the real world contribution of playthrough evaluation it would not be sufficient to stop there. Just as different evaluators treat candidate issues differently to one another, so too are developers likely to interpret and consider usability problems with different priorities to the evaluators. What's more, reviewers and players also rate games differently, as seen in the difference in ratings given by professional reviewers and private individual players on websites like Metacritic [1], for example. An extension to the studies presented in this thesis

[1] http://www.metacritic.com/

would be to explore the potential for playthrough evaluation to help guide development of a product from prototyping through to release and into the market. It would be useful to examine whether professional reviewers and players give the same feedback and interpretation as users in a lab environment, evaluators, and developers. Clearly this kind of longitudinal case study would require industry collaboration, and so remains as a possible suggestion for future work.

8.4.2 Usability-Playability, User Experience-Player Experience

The relationship between usability and playability needs to be understood and explored further, such as to understand under what conditions do usability issues become playability issues. For example, there are many examples of games with relatively poorly designed control systems that are still considered to be great games (e.g., *Grand Theft Auto*). This is likely to depend on an understanding of different kinds of players, their expertise, backgrounds, tastes, and competencies.

Having established a reliable method for usability evaluation, the stage is now set to expand the playthrough evaluation framework to explore issues of player experience. By exposing explicit criteria in usability patterns, the methodology has been demonstrated to facilitate introspection and analysis. This allowed evaluators' expertise to be examined, as well as identifying and addressing specific areas of weakness in the method where evaluator agreement was generally poor.

8.4.3 Extending "Interaction" to "Experience"

In order to extend the player action framework into a more complete *Player Experience Framework*, a hierarchical taxonomy of the components of experience would be needed, comparable to the user action framework. This is currently beyond the capability of the research field, as only very tentative, preliminary structures have been proposed. Further research is needed in order to first identify what these elements are. For example:

1. Explicating concepts and the measurement of such qualities as
    - Immersion
    - Presence
    - Engagement
    - Flow
2. How these components change with novel interfaces, or different contexts for example,
    - Game styles and genres
3. The contribution of novel ways to measure the player experience such as
    - Biometrics
4. Exploring components affecting aspects of player psychology,
    - Individual player differences, including:
        - Cognitive
        - Physical
        - Preference
        - Motivation

5. Unpacking more aggregated notions of
    - Satisfaction
    - Gameplay
    - Fun
6. The relationships between constructs,
    - Usability
    - Playability
    - User experience
    - Player experience

In order to develop a comparable taxonomy of player experience, all of these and a host of other issues need to be developed to the same extent that the research communities have done with usability over the last few decades.

Towards Affective Engineering

It is noteworthy that much of the theoretical underpinnings of the player action framework, by way of the user action framework and Norman's theory of action (Norman, 1986), derive from the early field of cognitive engineering. This emphasis on cognitive aspects of interaction is particularly relevant for a concern with usability. However, this thesis has delimited the boundaries for game evaluation, and pointed out those regions of player experience that are not well served by this kind of attention. There is burgeoning research in the areas of affective computing, gamification, and ludology, all of which may have important contributions to make.

8.4.4 Understanding Users Improves Evaluation

Hollnagel (1993a,b) presented a theory of phenotypes and genotypes which made a distinction between the causes of erroneous actions as being *system-induced* or *residual*. System-induced erroneous actions are considered to be due to characteristics of the interface, whereas residual erroneous actions are due to the variability and individual differences in cognitive and performance characteristics of users. This dichotomy is primarily a theoretical one, but it does point to an important factor. Many approaches to usability evaluation only consider errors from the perspective of system-inducement. i.e., they are primarily concerned with characteristics and design features in the interface. However, it is clear that design features that are good for one person or group of users may result in errors for another individual or group.

Improving the validity of the method would depend on the evaluators' understanding of users. Evaluators' opinion is usually based on implicit, subjective opinion rather than objective, empirical data. Chapter 2 (Literature Review) briefly described the concept of user effect, which refers to the differences in evaluation that depend on the user population chosen. Validity of the method could be improved by careful analysis of user data, making it explicit through transcription using the playthrough evaluation framework, rather than relying on individual evaluators' implicit and subjective opinions.

As the evaluator effect cannot be entirely removed it would be interesting to explore the inevitable differences in problem discovery and analysis between different evaluators. It could be possible to test evaluators' ability to predict real user behaviour, and so evaluate which

evaluators produce more valid results than others. This also has potential benefits for evaluator training, as benchmarks for evaluator performance could be generated based on real user test sessions. The evaluator's ability to discover and analyse a set of known problems could then be assessed, and areas for improvement identified.

8.4.5 Usability Remains Critical During Technological Innovation

Given the fast pace of development within the industry, it is expected that usability will remain a crucial issue in gaming, particularly as new technologies for control and display become ubiquitous. Even during the development of this thesis, several radical innovations have begun to break into the mainstream, including ubiquitous touch screens, 3D television, virtual reality headsets, augmented reality, full body interaction, and a boom in mobile gaming. Each of these comes with their own new forms of usability problem, which are exacerbated with their own particular specifics in the gaming context.

Furthermore, in recent years there has been a shift away from the more traditional, "hard-core" gaming demographic, towards a "casual" or mainstream audience. Successful development studios have made the most of this shift and have adapted the style of game to these new forms of audience, many of whom do not have a legacy of gaming expertise on which to draw. Many are unfamiliar with the traditional mores of gaming, and so there is a need to understand what gaming experiences causes basic usability problems for them.

Additionally, concomitant to the change in demographics, the games themselves are changing to be more suitable to their new audience. In some cases this means shorter durations, and in other, more supportive training or tutorials. Punishing learning curves which once were suitable for the determined, focussed player with plenty of time available, are now being softened so that the experience is rewarding for all skill levels throughout.

These changing circumstances are sufficient to occupy many research programmes in the mid-term future, and it is reasonable to expect the rate of change to continue. The player action framework is well suited to these changing times by being an extensible framework with clear procedures for testing and validation with novel data.

Appendices

Chapter Contents

Appendix A

Principal Component Analysis

A.1 Components

The following lists the 21 components identified through principal components analysis in Chapter 4 (Testing Heuristic Evaluation for Video Games). This list was produced by inspecting the components identified by principal components analysis for each of the 3 evaluators, and merging conceptually related groups together, similar to a closed card sort.

(1) Learning Skills (Controls, Mechanics, Tactics).

(2) Challenge.

(3) Manual & Tutorial (Help and Documentation).

(4) Usable Controls.

(5) Feedback.

(6) UI (HUD, Menu).

(7) Save Progress.

(8) Visibility (Visibility of System Status).

(9) Various Styles.

(10) Entertainment.

(11) Player in Control.

(12) Visual Representation Form & Function.

(13) Balance.

(14) Clear Goals.

(15) Unreasonable/Unexpected/Unacceptable Errors (Error Prevention).

(16) Approachability.

(17) External Consistency (Consistency and Standards).

(18) Audio/Visual Aesthetics.

(19) Internal Consistency.

(20) Screen Layout.

(21) Player Emotional Involvement With Character.

A.2 19 Components

The 21 components (Appendix A.1 - Components) were reduced to 19 by removing the following 2 due to lack of relevance with games in the thesis,

- Component 7: Save Progress
- Component 21: Player Emotional Involvement With Character

The remaining 19 components are as follows,

- Component 1: Learning Skills (Controls, Mechanics, Tactics)
- Component 2: Challenge
- Component 3: Manual & Tutorial (Help and Documentation)
- Component 4: Usable Controls
- Component 5: Feedback
- Component 6: UI (HUD, Menu)
- Component 8: Visibility (Visibility of System Status)
- Component 9: Various Styles
- Component 10: Entertainment
- Component 11: Player in Control
- Component 12: Visual Representation Form & Function
- Component 13: Balance
- Component 14: Clear Goals
- Component 15: Unreasonable/Unexpected/Unacceptable Errors (Error Prevention)
- Component 16: Approachability
- Component 17: External Consistency (Consistency and Standards)

- Component 18: Audio/Visual Aesthetics
- Component 19: Internal Consistency
- Component 20: Screen Layout

A.3 Scree Plots

Scree plots for each individual evaluator are shown in the following:

- Fig. A.1 ("Evaluator 1") on the current page.
- Fig. A.2 ("Evaluator 2") on the following page.
- Fig. A.3 ("Evaluator 3") on page 162.

The scree plot for the aggregate of all three evaluators' components is shown in:

- Fig. A.4 ("Aggregated") on page 163.

Figure A.1: Evaluator 1

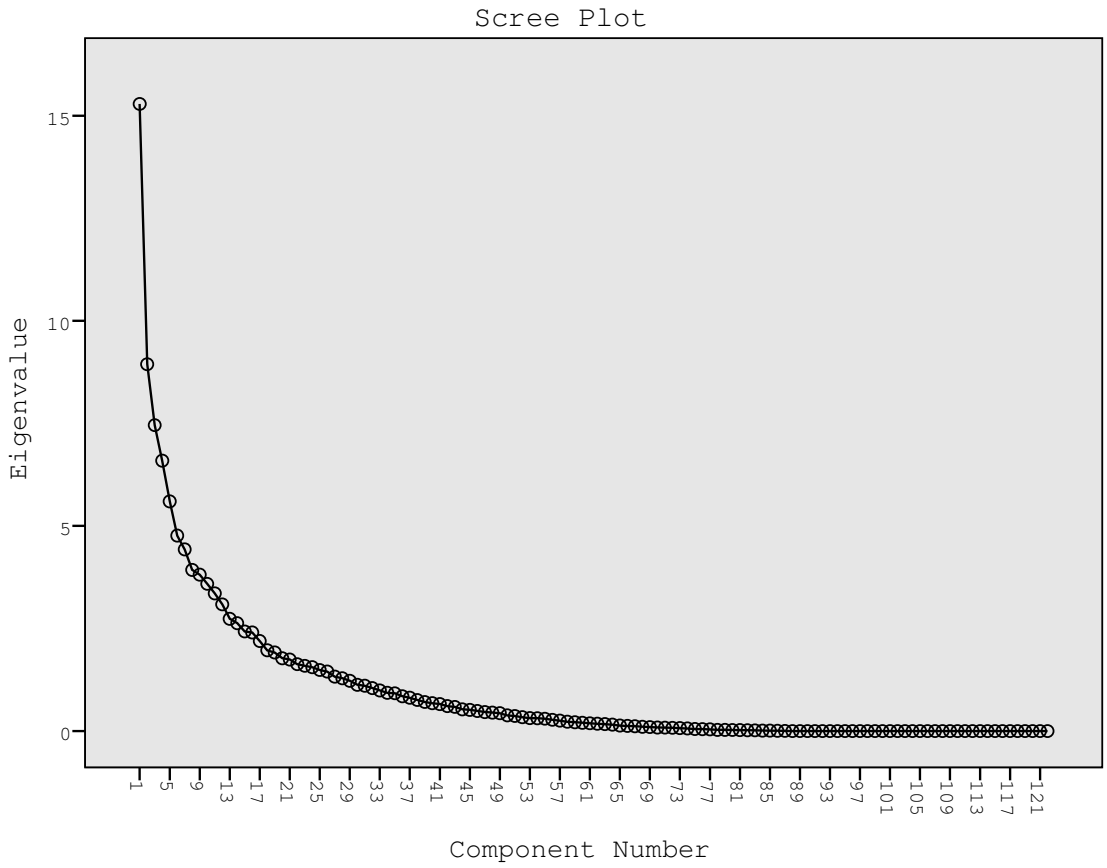


Figure A.2: Evaluator 2

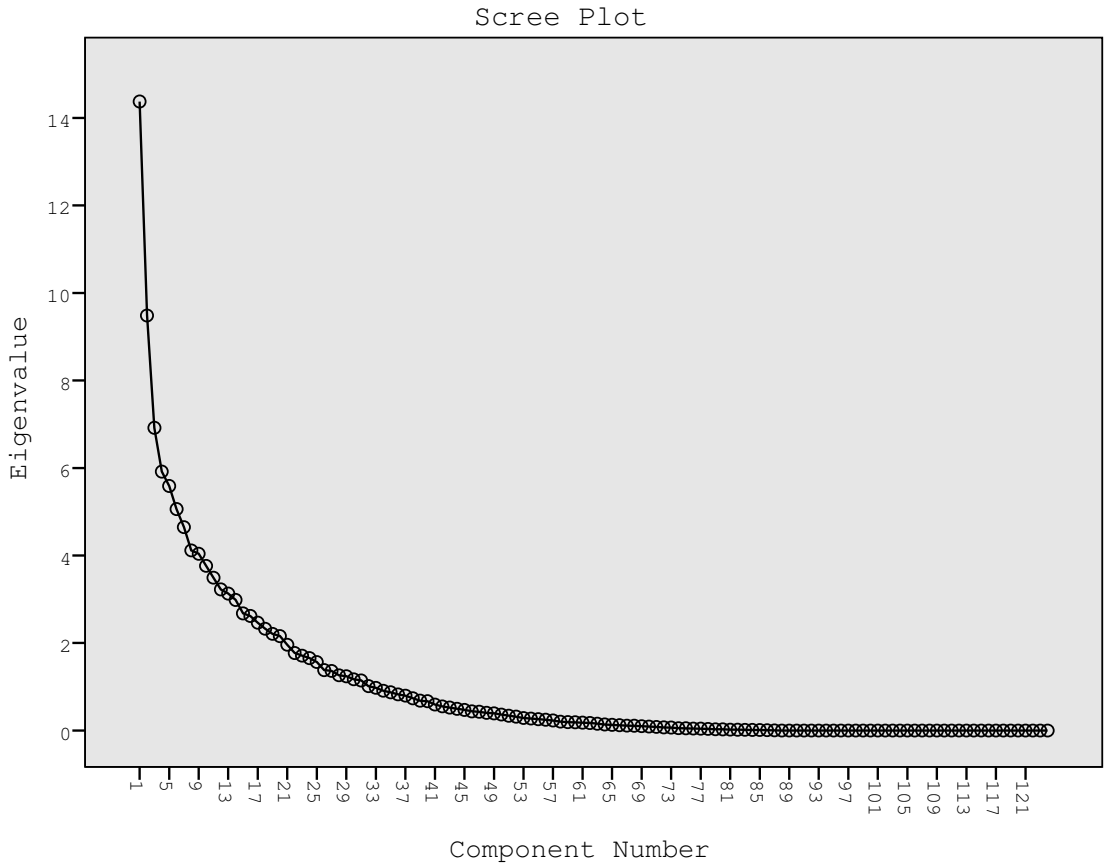


Figure A.3: Evaluator 3

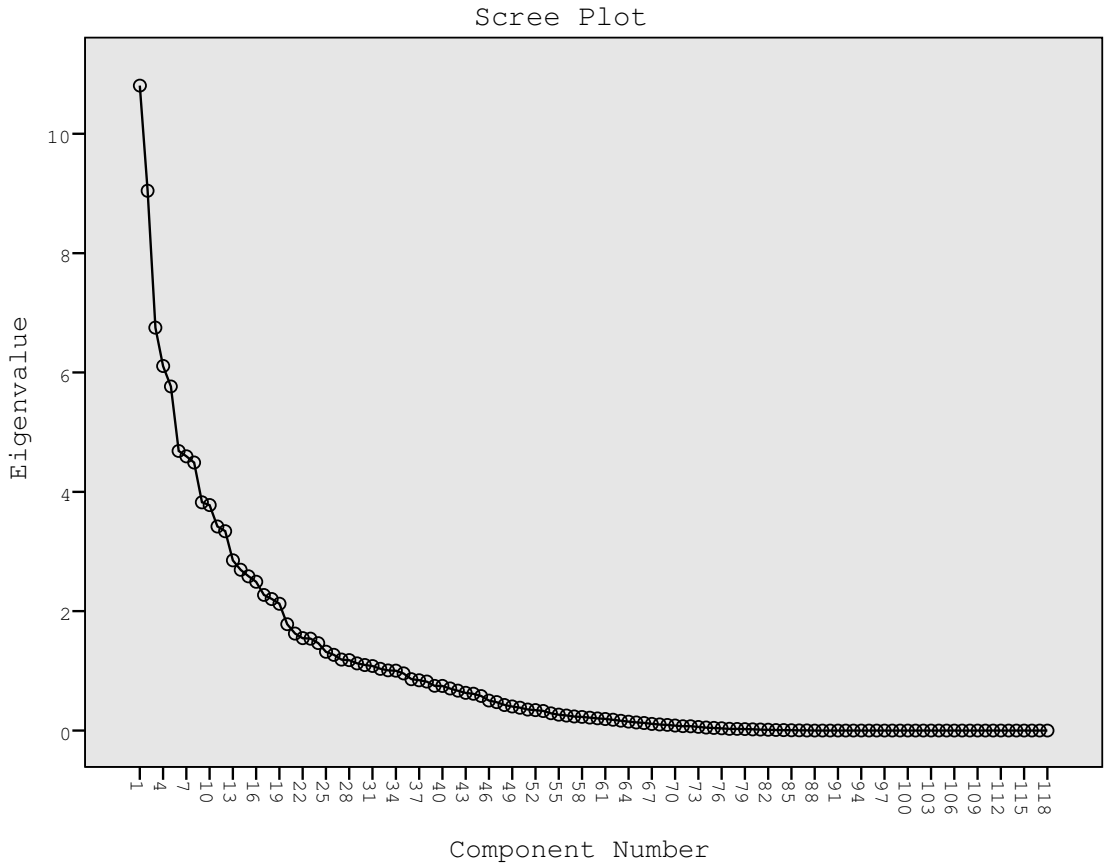
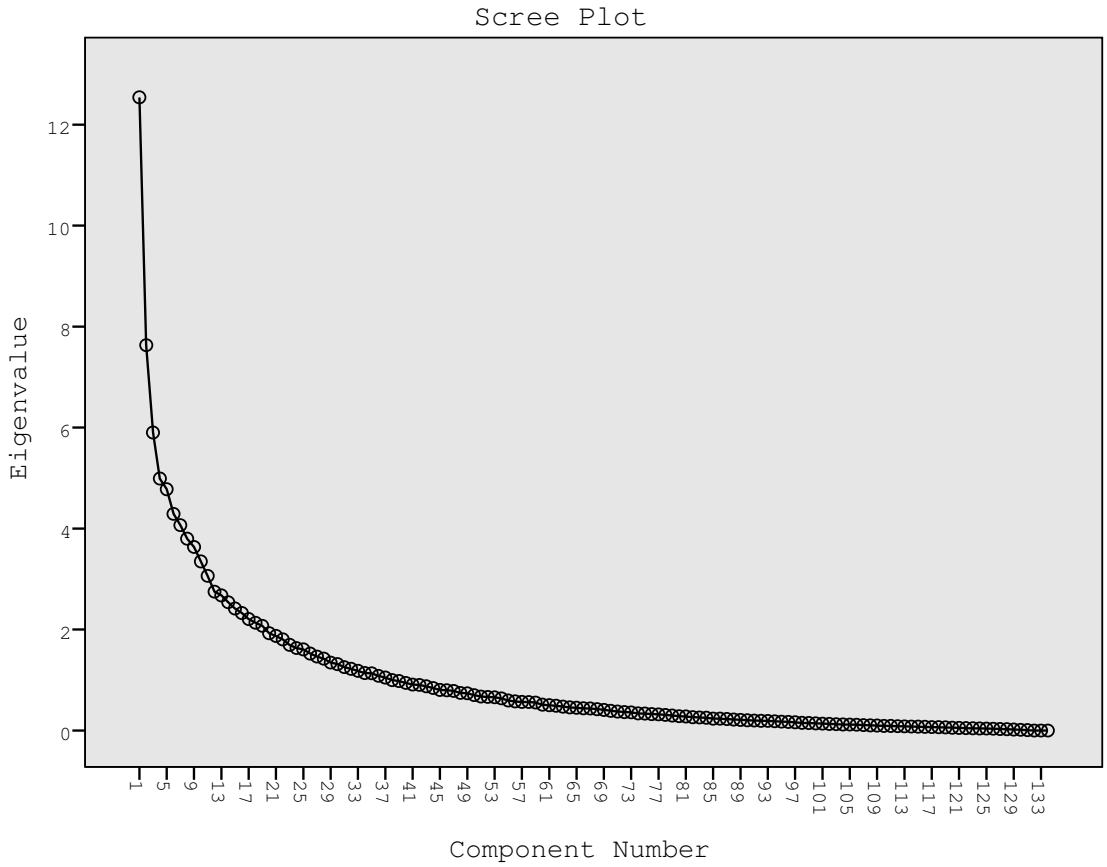


Figure A.4: Aggregated



A.4 Component Variance

The principal components identified for each individual evaluator are shown in the following:

- Table A.1 ("Evaluator 1 Principal Components") on the current page.
- Table A.2 ("Evaluator 2 Principal Components") on the following page.
- Table A.3 ("Evaluator 3 Principal Components") on page 166.

The principal components identified in the aggregate of all evaluators' ratings is shown in:

- Table A.4 ("Aggregated Principal Components") on page 167.

Table A.1: Evaluator 1 Principal Components

% Variance	Component
8.73	Learnability
5.68	Controls
5.35	Fun Challenge
4.97	Varied, Balanced Styles
3.9	Feedback
3.52	Usable HUD / Interface
3.41	Saveable Persistence
3.24	Ingame Help and Documentation
3.04	Appropriate Challenge and Pace
2.64	Recognisable Art
2.45	Audio / Visual Interest
2.24	Visible Status
2.09	Player in Control
1.66	Player Memory

Table A.2: Evaluator 2 Principal Components

% Variance	Component
7.69	Player Learning
7.33	Fun Through Balanced Challenge
4.94	Help and Documentation
4.74	Fun Through Non-Boring Progress
4.24	Visibility of Status & HUD
3.68	Controls
3.67	Feedback and Consistency
3.38	Player in Control
3.00	Varied and Balanced Styles
2.84	Balanced AI and Strategy
2.68	Unintentional Error
2.62	Audio / Visual Arousal
2.6	Understandable Visuals
2.37	Consistency
2.21	Help With Goals
2.03	Similar to Other Games
1.75	Interactive World
1.74	Clear Goals
1.72	Consistent Interface
1.7	Excessive Player Memory
1.75	Interactive World
1.74	Clear Goals
1.72	Consistent Interface
1.7	Excessive Player Memory
1.51	Interface
1.39	Screen
1.33	Difficulty
1.19	Controls

Table A.3: Evaluator 3 Principal Components

% Variance	Component
6.82	Balanced challenge
5.18	Controls
4.95	Player Interest and Engagement, Not Boredom
4.31	Balanced AI and Atrategy
4.05	Player Learning Skills
4.04	Help to Understand and Meet Goals
3.16	Unintentional Errors
3.1	Visibility of Status
2.96	Recognisable Art
2.91	Player Control and Learning
2.78	Efficient and Effective Screen Interface
2.53	Feedback / World Reaction
2.44	Feedback
2.4	Realworld Consistency
2.38	Mastery
2.31	Accessibility
2.23	Accessibility
1.95	Internal Consistency
1.94	Help and Documentation
1.75	No Manual Needed
1.5	Memory
1.37	Audio
1.34	Learning Curve
1.28	External Consistency

Table A.4: Aggregated Principal Components

% Variance	Component
6.11	Learning Skills (Controls, Mechanics, Tactics)
4.55	Challenge
4.43	Manual & Tutorial (Help and Documentation)
4.08	Usable Controls
3.32	Feedback
2.81	UI (HUD, Menu)
2.76	Save Progress
2.65	Visibility (Visibility of System Status)
2.63	Various Styles
2.45	Entertainment
2.43	Player In Control
2.41	Visual Representation Form & Function
2.29	Balance
2.25	(Conceptually) Clear Goals
2.05	Unreasonable/Unexpected/Unacceptable Errors (Error prevention)
1.97	Approachability
1.91	External Consistency (Consistency and Standards)
1.79	Audio/Visual
1.67	Internal Consistency
1.65	Screen Layout
1.64	Player Emotional Involvement With Character

A.5 Component Loadings

The heuristics loaded for the "Learning Skills" component are shown for each individual evaluator in the following:

- Table A.5 ("Evaluator 1 Learning Skills heuristic loadings") on the next page.
- Table A.6 ("Evaluator 2 Learning Skills heuristic loadings") on page 170.
- Table A.7 ("Evaluator 3 Learning Skills heuristic loadings") on page 170.

The heuristics loaded to the "Learning Skills" component for the aggregate of all three evaluators' components is shown in:

- Table A.8 ("Aggregated Learning Skills heuristic loadings") on page 171.

Table A.5: Evaluator 1 Learning Skills heuristic loadings

Loading	Heuristic
0.84	Heuristic 95: "Teach skills early that you expect the players to use later"
0.83	Heuristic 68: "Player given opportunity to model correct behavior and skills"
0.83	Heuristic 74: "Player provided with opportunities to practice new skills so as to commit skills to memory"
0.80	Heuristic 136: "The skills needed to attain goals are taught early enough to play or use later, or right before the new skill is needed"
0.80	Heuristic 60: "Player able to succeed at playing game after training period, i.e., first level or tutorial"
0.78	Heuristic 57: "Player able to demonstrate and practice new actions without severe consequences. Player knows what actions to take"
0.76	Heuristic 79: "Provide an interesting and absorbing tutorial"
0.74	Heuristic 47: "Learning curve is too steep; requires too much micromanagement; command sequences are complex, lengthy, and awkward, making the game difficult to play"
0.69	Heuristic 28: "Easy to learn, harder to master"
0.61	Heuristic 83: "Provide instructions, training, and help"
0.61	Heuristic 78: "Players should be given context sensitive help while playing so that they are not stuck and need to rely on a manual for help"
0.61	Heuristic 1: "A good game should be easy to learn and hard to master (Nolan Bushnell)"
0.60	Heuristic 20: "Consistency shortens the learning curve by following the trends set by the gaming industry to meet users' expectations. If no industry standard exists, perform usability / playability research to ascertain the best mapping for the majority of intended players"
0.53	Heuristic 72: "Player is given controls that are basic enough to learn quickly, yet expandable for advanced options for advanced players"
0.44	Heuristic 3: "Actions and skills learned were important for playing the game not just for a single event in the game"
-0.41	Heuristic 91: "Screen layout is efficient and visually pleasing"
0.40	Heuristic 6: "All levels of players are able to play and get involved quickly and easily with tutorials, and/or progressive or adjustable difficulty levels"

Table A.6: Evaluator 2 Learning Skills heuristic loadings

Loading	Heuristic
0.88	Heuristic 68: "Player given opportunity to model correct behavior and skills"
0.87	Heuristic 74: "Player provided with opportunities to practice new skills so as to commit skills to memory"
0.84	Heuristic 136: "The skills needed to attain goals are taught early enough to play or use later, or right before the new skill is needed"
0.80	Heuristic 57: "Player able to demonstrate and practice new actions without severe consequences. Player knows what actions to take"
0.77	Heuristic 95: "Teach skills early that you expect the players to use later"
0.72	Heuristic 1: "A good game should be easy to learn and hard to master (Nolan Bushnell)"
0.70	Heuristic 72: "Player is given controls that are basic enough to learn quickly, yet expandable for advanced options for advanced players"
0.65	Heuristic 60: "Player able to succeed at playing game after training period, i.e., first level or tutorial"
0.61	Heuristic 40: "Get the player involved quickly and easily"
0.60	Heuristic 6: "All levels of players are able to play and get involved quickly and easily with tutorials, and/or progressive or adjustable difficulty levels"
0.50	Heuristic 58: "Player able to master game using skills and tools provided"

Table A.7: Evaluator 3 Learning Skills heuristic loadings

Loading	Heuristic
0.96	Heuristic 60: "Player able to succeed at playing game after training period, i.e., first level or tutorial"
0.93	Heuristic 47: "Learning curve is too steep; requires too much micromanagement; command sequences are complex, lengthy, and awkward, making the game difficult to play"
0.86	Heuristic 68: "Player given opportunity to model correct behavior and skills"
0.84	Heuristic 57: "Player able to demonstrate and practice new actions without severe consequences. Player knows what actions to take"
0.83	Heuristic 95: "Teach skills early that you expect the players to use later"
0.83	Heuristic 1: "A good game should be easy to learn and hard to master (Nolan Bushnell)"
0.76	Heuristic 136: "The skills needed to attain goals are taught early enough to play or use later, or right before the new skill is needed"
0.75	Heuristic 74: "Player provided with opportunities to practice new skills so as to commit skills to memory"
0.53	Heuristic 63: "Player does not need to access the tutorial in order to play"

Table A.8: Aggregated Learning Skills heuristic loadings

Loading	Heuristic
0.84	Heuristic 95: "Teach skills early that you expect the players to use later"
0.82	Heuristic 68: "Player given opportunity to model correct behavior and skills"
0.82	Heuristic 136: "The skills needed to attain goals are taught early enough to play or use later, or right before the new skill is needed"
0.82	Heuristic 57: "Player able to demonstrate and practice new actions without severe consequences. Player knows what actions to take"
0.81	Heuristic 74: "Player provided with opportunities to practice new skills so as to commit skills to memory"
0.72	Heuristic 47: "Learning curve is too steep; requires too much micromanagement; command sequences are complex, lengthy, and awkward, making the game difficult to play"
0.71	Heuristic 28: "Easy to learn, harder to master"
0.68	Heuristic 79: "Provide an interesting and absorbing tutorial"
0.67	Heuristic 60: "Player able to succeed at playing game after training period, i.e., first level or tutorial"
0.64	Heuristic 1: "A good game should be easy to learn and hard to master (Nolan Bushnell)"
0.48	Heuristic 20: "Consistency shortens the learning curve by following the trends set by the gaming industry to meet users' expectations. If no industry standard exists, perform usability / playability research to ascertain the best mapping for the majority of intended players"
0.44	Heuristic 3: "Actions and skills learned were important for playing the game not just for a single event in the game"

Appendix B

Player Action Framework

B.1 Player Action Framework Events

This section of the appendix lists the event codes used in the player action framework. Each event is categorised according to type and interaction stage.

Types:

- Skill, Control, Mechanic, Feature, Action.
- Help.
- Goal, Task.
- UI, Feedback, Art, Audio.
- Usability Outcome.

Interaction Stages:

- Context.
- Breakdown.
- Outcome.

Event Summaries

(1) Skill/control/action/mechanic/feature/tactic (IS/NOT) introduced.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Context.

(2) Player (DOES/NOT) notice/recognise/understand skill/control/action/mechanic/feature/tactic explanation/introduction/presentation/tuition.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(3) Player (IS/NOT) provided with opportunity to practice skill/control/action/mechanic/feature/tactic.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Context.

(4) Player (DOES/NOT) practice/demonstrate necessary/desirable/appropriate/expected/correct competence with Skill/control/action/mechanic/feature/tactic.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(5) Moderator intervention: Explains/teaches/demonstrates/applies skill/control/action/mechanic/feature/tactic.

Type: Help.
Interaction stage: Outcome.

(6) Skill/control/action/mechanic/feature/tactic (UN/necessary/desirable/expected) for goal/task.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Context.

(7) (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic use (IS/NOT) attempted.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(8) (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic used (UN/SUCCESSFULLY).

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(9) (UN/necessary, UN/desirable, UN/expected or IN/correct) skill/control/action/mechanic/feature/tactic use attempted.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(12) Player (UN/AWARE) of existence of necessary/desirable/expected skill/control/action/mechanic/feature/tactic.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(13) How to use necessary/desirable/appropriate/correct/expected skill/control/action/mechanic/feature/tactic (IS/NOT) committed to (cognitive/muscle) memory.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(14) Effectiveness: Task (SUCCEEDED/FAILED, IN/COMPLETE).

Type: Usability Outcome.
Interaction stage: Outcome.

(15) Efficiency: Resource expenditure too (HIGH/LOW).

Type: Usability Outcome.
Interaction stage: Outcome.

(16) (DIS/SATISFACTION).

Type: Usability Outcome.
Interaction stage: Outcome.

(17) Player needs help/information/training/support mechanism.

Type: Help.
Interaction stage: Outcome.

(18) New goal/task set (IMPLICITLY/EXPLICITLY).

Type: Goal, Task.
Interaction stage: Context.

(19) Player (DOES/NOT) notice/recognise presentation of goal/task.

Type: Goal, Task.
Interaction stage: Breakdown.

(20) Player (DOES/NOT) understand goal/task.

Type: Goal, Task.
Interaction stage: Breakdown.

(21) Player has forgotten goal/task.

Type: Goal, Task.
Interaction stage: Breakdown.

(22) Player (DOES/NOT) understand function/purpose/effect/consequence of feature/skill/control/mechanic.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(23) Player (DOES/NOT) understand how to use control/feature/skill/mechanic.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(24) Control/feature/skill/mechanic (DOES/NOT) default/conform to industry standard.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Context.

(25) Control/mechanic/feature/action/interface (IS/NOT) consistent.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Context.

(26) Player (DOES/NOT) notice necessary/desirable UI/art/indicator.

Type UI, Feedback, Art, Audio.
Interaction stage: Breakdown.

(27) Player (DOES/NOT) understand/recognise purpose/meaning of UI/art/indicator.

Type UI, Feedback, Art, Audio.
Interaction stage: Breakdown.

(28) (UN/necessary/desirable) UI/art/indicator (IS/NOT) visible.

Type UI, Feedback, Art, Audio.
Interaction stage: Context.

(29) (UN/necessary/desirable/expected) feedback (DOES/NOT) occur.

Type UI, Feedback, Art, Audio.
Interaction stage: Context.

(30) Player (DOES/NOT) notice/recognise necessary/desirable/expected feedback.

Type UI, Feedback, Art, Audio.
Interaction stage: Breakdown.

(31) Player (DOES/NOT) understand feedback.

Type UI, Feedback, Art, Audio.
Interaction stage: Breakdown.

(32) Action/control/mechanic considered necessary/desirable/correct by player (IS/NOT) possible/allowed/appropriate/supported/reasonable.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(33) Player does not want to use mechanic/skill/feature/control/action.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(34) Player (DOES/NOT) understand presentation of goal/task.

Type: Goal, Task.
Interaction stage: Breakdown.

(35) Control/feature/skill/control/mechanic (DOES/NOT) behave/feel correct/natural.

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(36) Audio (DOES/NOT) feel correct/natural/realistic/as expected.

Type UI, Feedback, Art, Audio.
Interaction stage: Breakdown.

(37) Player (DOES/NOT) recognise that task/goal (SUCCEEDED/FAILED).

Type: Goal, Task.
Interaction stage: Breakdown.

(38) Player (DOES/NOT) understand why task/goal (SUCCEEDED/FAILED).

Type: Goal, Task.
Interaction stage: Breakdown.

(39) Player (DOES/NOT) recognise that action/control/skill/feature/mechanic (SUCCEEDED/FAILED).

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(40) Player (DOES/NOT) understand why action/control/skill/feature/mechanic (SUCCEEDED/FAILED).

Type: Skill, Control, Mechanic, Feature, Action.
Interaction stage: Breakdown.

(41) Style/approach/tactic/goal/task considered necessary/desirable/correct by player (IS/NOT) possible/allowed/appropriate/supported/reasonable.

Type: Goal, Task.
Interaction stage: Breakdown.

(42) Player does not want to use style/approach/tactic/goal/task.

Type: Goal, Task.
Interaction stage: Breakdown.

B.2 Player Action Framework Tree

This section presents the complete player action framework including Appendix A.1 (Components), their associated issues, and the relevant events, separated by interaction stage.

In most cases only a single heuristic and issue were used in the decomposition stage of Chapter 6 (The Playthrough Evaluation Framework). In cases where more than one issue was given the highest rating, all such candidates are were included in the decomposition and are reported here. Further discussion about the process and results of this derivation process are described in detail in Chapter 6 (The Playthrough Evaluation Framework).

1. Component 1: Learning Skills (Controls, Mechanics, Tactics)
    - Heuristic 74: "Player provided with opportunities to practice new skills so as to commit skills to memory"
    - Issue 59: "Player comments 'How do I attack him.'"
    - Issue 65: "Player comments 'OK I've forgotten which button it said I should press to go on to the ceiling or whatever.'"
    - Context:
        - Event 1: Skill/control/action/mechanic/feature/tactic (IS/NOT) introduced
        - Event 3: Player (IS/NOT) provided with opportunity to practice skill/control/action/mechanic/feature/tactic
    - Breakdown:
        - Event 2: Player (DOES/NOT) notice/recognise/understand skill/control/action/mechanic/feature/tactic explanation/introduction/presentation/tuition
        - Event 4: Player (DOES/NOT) practice/demonstrate necessary/desirable/appropriate/expected/correct competence with Skill/control/action/mechanic/feature/-tactic
    - Context:
        - Event 18: New goal/task set (IMPLICITLY/EXPLICITLY)
        - Event 6: Skill/control/action/mechanic/feature/tactic (UN/necessary/desirable/-expected) for goal/task
    - Breakdown:
        - Event 7: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic use (IS/NOT) attempted
        - Event 8: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic used (UN/SUCCESSFULLY)
        - Event 13: How to use necessary/desirable/appropriate/correct/expected skill/-control/action/mechanic/feature/tactic (IS/NOT) committed to (cognitive/muscle) memory
    - Outcome:
        - Event 14: Effectiveness: Task (SUCCEEDED/FAILED, IN/COMPLETE)
        - Event 15: Efficiency: Resource expenditure too (HIGH/LOW)
        - Event 16: (DIS/SATISFACTION)
2. Component 2: Challenge
    - Heuristic 16: "Challenge, strategy, and pace are in balance"
    - Issue 9: "Player comments that 'This point seems to have too many aliens in it.'"
    - Context:
        - Event 3: Player (IS/NOT) provided with opportunity to practice skill/control/action/mechanic/feature/tactic

- Breakdown:
  - Event 4: Player (DOES/NOT) practice/demonstrate necessary/desirable/appropriate/expected/correct competence with Skill/control/action/mechanic/feature/-tactic
- Context:
  - Event 18: New goal/task set (IMPLICITLY/EXPLICITLY)
  - Event 6: Skill/control/action/mechanic/feature/tactic (UN/necessary/desirable/-expected) for goal/task
- Breakdown:
  - Event 7: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic use (IS/NOT) attempted
  - Event 8: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic used (UN/SUCCESSFULLY)
- Outcome:
  - Event 14: Effectiveness: Task (SUCCEEDED/FAILED, IN/COMPLETE)
  - Event 15: Efficiency: Resource expenditure too (HIGH/LOW)
  - Event 16: (DIS/SATISFACTION)

3. Component 13: Balance
   - Heuristic 35: "Game play should be balanced so that there is no definite way to win"
   - Issue 48: "Predator seems very easy to kill if the player has the mini-gun"
   - Context:
     - Event 18: New goal/task set (IMPLICITLY/EXPLICITLY)
   - Outcome:
     - Event 14: Effectiveness: Task (SUCCEEDED/FAILED, IN/COMPLETE)
     - Event 15: Efficiency: Resource expenditure too (HIGH/LOW)
     - Event 16: (DIS/SATISFACTION)

4. Component 3: Manual & Tutorial (Help and Documentation)
   - Heuristic 83: "Provide instructions, training, and help"
   - Issue 40: "Moderator gives tutorial on how to use the alien"
   - Context:
     - Event 18: New goal/task set (IMPLICITLY/EXPLICITLY)
     - Event 1: Skill/control/action/mechanic/feature/tactic (IS/NOT) introduced
     - Event 3: Player (IS/NOT) provided with opportunity to practice skill/control/action/mechanic/feature/tactic
     - Event 4: Player (DOES/NOT) practice/demonstrate necessary/desirable/appropriate/expected/correct competence with Skill/control/action/mechanic/feature/-tactic
     - Event 13: How to use necessary/desirable/appropriate/correct/expected skill/-control/action/mechanic/feature/tactic (IS/NOT) committed to (cognitive/muscle) memory
   - Breakdown:
     - Event 7: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic use (IS/NOT) attempted
     - Event 8: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic used (UN/SUCCESSFULLY)
     - Event 9: (UN/necessary, UN/desirable, UN/expected or IN/correct) skill/control/action/mechanic/feature/tactic use attempted
   - Outcome:
     - Event 14: Effectiveness: Task (SUCCEEDED/FAILED, IN/COMPLETE)

  - Event 15: Efficiency: Resource expenditure too (HIGH/LOW)
  - Event 17: Player needs help/information/training/support mechanism
  - Event 16: (DIS/SATISFACTION)
  - Event 5: Moderator intervention: Explains/teaches/demonstrates/applies skill/control/action/mechanic/feature/tactic

5. Component 10: Entertainment
   - Heuristic 133: "The players finds the game fun, with no repetitive or boring tasks"
   - Issue 64: "Player comments that the hacking scene takes a bit long, seems drawn out"
   - Outcome:
     - Event 16: (DIS/SATISFACTION)

6. Component 6: UI (HUD, Menu)
   - Heuristic 46: "Interfaces should be consistent in control, color, typography, and dialog design"
   - Issue 37: "Player seems lost, no waypoint indicator at top of screen"
   - Context:
     - Event 25: Control/mechanic/feature/action/interface (IS/NOT) consistent

7. Component 8: Visibility (Visibility of System Status)
   - Heuristic 45: "Indicators are visible"
   - Issue 29: "Player comments that 'They should highlight it more when you're about to run out of energy.'"
   - Context:
     - Event 28: (UN/necessary/desirable) UI/art/indicator (IS/NOT) visible
     - Event 29: (UN/necessary/desirable/expected) feedback (DOES/NOT) occur
   - Breakdown:
     - Event 26: Player (DOES/NOT) notice necessary/desirable UI/art/indicator
     - Event 27: Player (DOES/NOT) understand/recognise purpose/meaning of UI/art/indicator
     - Event 30: Player (DOES/NOT) notice/recognise necessary/desirable/expected feedback
     - Event 31: Player (DOES/NOT) understand feedback

8. Component 20: Screen Layout
   - Heuristic 92: "Screen layout is efficient, integrated, and visually pleasing"
   - Issue 16: "There are three separate textual messages on the screen simultaneously"
   - Breakdown:
     - Event 26: Player (DOES/NOT) notice necessary/desirable UI/art/indicator
     - Event 27: Player (DOES/NOT) understand/recognise purpose/meaning of UI/art/indicator
   - Context:
     - Event 28: (UN/necessary/desirable) UI/art/indicator (IS/NOT) visible
   - Outcome:
     - Event 16: (DIS/SATISFACTION)

9. Component 4: Usable Controls
   - Heuristic 23: "Controls should be intuitive and mapped in a natural way"
   - Issue 12: "Player comments 'The button controls aren't entirely intuitive', also block on the left bumper can be a little frustrating to use"
   - Context:

    - Event 24: Control/feature/skill/mechanic (DOES/NOT) default/conform to industry standard
  - Breakdown:
    - Event 22: Player (DOES/NOT) understand function/purpose/effect/consequence of feature/skill/control/mechanic
    - Event 23: Player (DOES/NOT) understand how to use control/feature/skill/mechanic
    - Event 7: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic use (IS/NOT) attempted
    - Event 8: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic used (UN/SUCCESSFULLY)
  - Outcome:
    - Event 16: (DIS/SATISFACTION)
    - Event 15: Efficiency: Resource expenditure too (HIGH/LOW)
    - Event 14: Effectiveness: Task (SUCCEEDED/FAILED, IN/COMPLETE)

10. Component 5: Feedback
    - Heuristic 80: "Provide appropriate audio/visual/visceral feedback (music, sound effects, controller vibration)"
    - Issue 52: "Player says 'Does that beep mean that I've scanned him?'"
    - Context
      - Event 29: (UN/necessary/desirable/expected) feedback (DOES/NOT) occur
    - Breakdown:
      - Event 30: Player (DOES/NOT) notice/recognise necessary/desirable/expected feedback Event 31: Player (DOES/NOT) understand feedback
    - Outcome:
      - Event 16: (DIS/SATISFACTION)

11. Component 11: Player in Control
    - Heuristic 76: "Players feel in control"
    - Issue 56: "Player starts charging his energy, but gets attacked at the same time. He comments 'Maybe some way to cancel that would be useful.'"
    - Breakdown:
      - Event 32: Action/control/mechanic considered necessary/desirable/correct by player (IS/NOT) possible/allowed/appropriate/supported/reasonable
      - Event 33: Player does not want to use mechanic/skill/feature/control/action
      - Event 41: Style/approach/tactic/goal/task considered necessary/desirable/correct by player (IS/NOT) possible/allowed/appropriate/supported/reasonable

12. Component 9: Various Styles
    - Heuristic 116: "The game supports different playing styles"
    - Issue 4: "Player says 'It's very difficult to sneak up on people, going slowly etc.' He says his next approach is 'I'll rush in an use the laptop before they can kill me.'"
    - Breakdown:
      - Event 32: Action/control/mechanic considered necessary/desirable/correct by player (IS/NOT) possible/allowed/appropriate/supported/reasonable
      - Event 33: Player does not want to use mechanic/skill/feature/control/action
      - Event 41: Style/approach/tactic/goal/task considered necessary/desirable/correct by player (IS/NOT) possible/allowed/appropriate/supported/reasonable
    - Outcome:
      - Event 16: (DIS/SATISFACTION)

13. Component 15: Unreasonable/Unexpected/Unacceptable Errors (Error Prevention)
    - Heuristic 42: "Help users recognize, diagnose, and recover from errors"
    - Issue 31: "Player comments that 'Sometimes I can't eat the brains of some people that I kill to restore my health, that's a bit annoying.'"
    - Breakdown:
      - Event 39: Player (DOES/NOT) recognise that action/control/skill/feature/mechanic (SUCCEEDED/FAILED)
      - Event 40: Player (DOES/NOT) understand why action/control/skill/feature/mechanic (SUCCEEDED/FAILED)
      - Event 37: Player (DOES/NOT) recognise that task/goal (SUCCEEDED/FAILED)
      - Event 38: Player (DOES/NOT) understand why task/goal (SUCCEEDED/FAILED)
14. Component 18: Audio/Visual Aesthetics
    - Heuristic 119: "The game utilizes visceral, audio and visual content to further the players' immersion in the game"
    - Issue 44: "Player comments that the shotgun sounds a bit wimpy at the moment"
    - Outcome:
      - Event 16: (DIS/SATISFACTION)
15. Component 12: Visual Representation Form & Function
    - Heuristic 11: "Art is recognizable to the player and speaks to its function"
    - Issue 3: "Player tries many times to jump off from part of the level, the graphics suggest that he should be allowed to. Other players have tried this also"
    - Breakdown:
      - Event 27: Player (DOES/NOT) understand/recognise purpose/meaning of UI/art/indicator
16. Component 17: External Consistency (Consistency and Standards)
    - Heuristic 50: "Mechanics should feel natural and have correct weight and momentum"
    - Issue 75: "Player comments that the speed of the scope zoom is a little bit slow, speed it up"
    - Breakdown:
      - Event 35: Control/feature/skill/control/mechanic (DOES/NOT) behave/feel correct/natural
    - Outcome:
      - Event 16: (DIS/SATISFACTION)
17. Component 19: Internal Consistency
    - Heuristic 19: "Consistency and Standards"
    - Issue 37: "Player seems lost, no waypoint indicator at top of screen"
    - Context:
      - Event 24: Control/feature/skill/mechanic (DOES/NOT) default/conform to industry standard
      - Event 25: Control/mechanic/feature/action/interface (IS/NOT) consistent
    - Outcome:
      - Event 16: (DIS/SATISFACTION)
18. Component 14: Clear Goals
    - Heuristic 104: "The game goals are clear. The game provides clear goals, presents overriding goals early as well as short term goals throughout game play"

- Issue 25: "Player checks in-game options after playing for 10 minutes, possibly looking for help or a map. Moderator intervenes to remind him of the objectives screen"
- Context:
  - Event 18: New goal/task set (IMPLICITLY/EXPLICITLY)
- Breakdown:
  - Event 19: Player (DOES/NOT) notice/recognise presentation of goal/task
  - Event 34: Player (DOES/NOT) understand presentation of goal/task
  - Event 20: Player (DOES/NOT) understand goal/task
  - Event 21: Player has forgotten goal/task
- Outcome:
  - Event 14: Effectiveness: Task (SUCCEEDED/FAILED, IN/COMPLETE)
  - Event 15: Efficiency: Resource expenditure too (HIGH/LOW)
  - Event 16: (DIS/SATISFACTION)

19. Component 16: Approachability
    - Heuristic 58: "Player able to master game using skills and tools provided"
    - Issue 20: "Player comments about the marines that it's difficult to 'get a hold of these people and kill them.'"
    - Breakdown:
      - Event 2: Player (DOES/NOT) notice/recognise/understand skill/control/action/mechanic/feature/tactic explanation/introduction/presentation/tuition
      - Event 22: Player (DOES/NOT) understand function/purpose/effect/consequence of feature/skill/control/mechanic
      - Event 23: Player (DOES/NOT) understand how to use control/feature/skill/mechanic
    - Heuristic 40: "Get the player involved quickly and easily"
    - Outcome:
      - Event 16: (DIS/SATISFACTION)
    - Heuristic 98: "The first-time experience is encouraging"
    - Issue 9: "Player comments that 'This point seems to have too many aliens in it.'"
    - Outcome:
      - Event 16: (DIS/SATISFACTION)
    - Issue 57: "Player comments that 'Every time I do a finishing move, he just pulls away every time. Not sure I want to use a finishing move.'"
    - Breakdown:
      - Event 33: Player does not want to use mechanic/skill/feature/control/action
    - Outcome:
      - Event 16: (DIS/SATISFACTION)

20. Component 7: Save Progress
    - Heuristic 124: "The player does not lose any hard-won possessions"

21. Component 21: Player Emotional Involvement With Character
    - Heuristic 70: "Player identifies with character"

Note that the final 2 from this list, Component 7: Save Progress and Component 21: Player Emotional Involvement With Character were excluded from further analysis due to lack of relevance to the games evaluated in this thesis. Future work will consider how to expand the player action framework tree to accommodate further events applicable to other genres and styles of game.

Appendix C

Study Materials

C.1 Playthough Evaluation Procedure

This section introduces the playthrough evaluation method and gives instructions to evaluators for how to evaluate video games using it.

C.1.1 Terminology

"Component"
: Components are the general areas that are considered when analysing usability issues.
Each component consists of a short name indicating the topic, example heuristics that describe relevant design guidelines, and interaction scenarios which represent the typical sequence of interaction.

"Heuristic"
: A heuristic is a simple rule of thumb or guideline that describes common properties of a systems with good usability.

"Event"
: Events are single, discrete parts of interaction scenarios that occur during gameplay.
During evaluation they are the criteria to be evaluated for each issue analysed.
They are often formatted to include either/or conditions such as whether a particular feature was / was not used by the player. During evaluation the evaluator specifies whether the conditions were or were not met.
Additionally they may include a number of similar aspects of the game that the event can be applicable to.
For example, Event 1: Skill/control/action/mechanic/feature/tactic (IS/NOT) introduced.
In this case if any of the optional components (skill, control, action, mechanic, feature, or tactic) is NOT introduced where it should have been then this event would be violated.

"Interaction scenario"
: A series of events that describe a typical interaction for a single component.
The evaluator considers each of the events listed and states whether they have occurred or not.
Scenarios are structured into different stages: context, breakdown, and outcome.
Scenarios consist of one or more of these interaction stages, and often there will be one of each, listed in this order.

Not all of the stages are required, though before an Outcome occurs, a Breakdown must be defined.

"Context"
: The stage of an interaction scenario that lists events that precede a problem, that the subsequent events depend on.
For example, Event 1: Skill/control/action/mechanic/feature/tactic (IS/NOT) introduced.
This event is evaluated when one or more of the components listed (skill, control, action, etc) is expected to be introduced. Even if the component were not introduced at this stage a breakdown would not necessarily occur as the player may still be able to successfully play the game without it.

"Breakdown"
: The stage of an interaction scenario that lists the events where a problem potentially occurs.
For example, Event 2: Player (DOES/NOT) notice/recognise/understand skill/control/action/mechanic/feature/tactic explanation/introduction/presentation/tuition
In this case the event describes a breakdown if the player does not either notice, recognise, or understand what has been explained, introduced, presented, or taught. If the player successfully noticed, recognised, and understood then the event is conformed to and no breakdown actually occurred.

"Outcome"
: The stage of an interaction scenario that lists the usability consequences after the problem has occurred. Most of the outcomes are defined in terms of traditional aspects of usability including efficiency, effectiveness, and satisfaction.
For example, Event 14: Effectiveness: Task (SUCCEEDED/FAILED, IN/COMPLETE).
If a task failed, or was incomplete when it should have already been successful then this event describes an violation of the effectiveness outcome.

C.1.2 Overview

Using playthrough evaluation involves the following steps:

1. Training.
2. Game play.
3. Issue detection.
4. Issue analysis.

Each of these stages is described in further detail in the following sections.

C.1.3 Training

Review the event codes, interaction scenarios, heuristics, and components in the tree hierarchy: Appendix B.2 (Player Action Framework Tree). Read the rest of this document to familiarise yourself with the overall purpose and method. If there are any points that are unclear, please ask the facilitator for clarification.

C.1.4 Game Play

Play the game in a natural way as you would without any particular evaluation in mind. The purpose of this stage is to get a natural feeling for how the game plays, and to immerse yourself into the player experience. The only additional activity you should do is to think out loud whenever possible. Describe what you're doing in the game, what you think or feel about any aspect that comes to mind. For example, if you like or dislike something, if you're uncertain or feel confident that you know what you have to do next, if you understand or are unclear about anything. The game play session will be recorded, and in the subsequent evaluation stage you will review your own footage and apply the playthrough evaluation method.

It's important to bear in mind that *you* are not being tested in any way. However you play the game, and whatever you think about it is completely fine and treated as confidential. Please be as honest as direct as possible. Your experience, opinions and comments whether positive or negative are very valuable.

Issue Detection

This is the start of the evaluation itself. Play back the video of the game session from the beginning. Look for the following cases:

- An actual problem occurs that affects the gameplay in the footage.
- A possible problem occurs that does not affect the gameplay in the footage, but which you predict could potentially affect other players.

Candidate problems are where play does not proceed in the correct, preferred, expected, or optimal manner. For example, the player makes a mistake, or expresses (verbally or non-verbally) something negative about their game experience.

Whenever you observe either of these cases, pause the video and review all of your previously recorded issues. Has the current issue already been recorded earlier?

- Yes:
  Add the new timestamp (an instant or a duration that the incident occurred over) to the previous report.
  Add additional events if necessary (e.g., count multiple failures) according to the Candidate Analysis procedure following.
- No:
  Create a new report following the procedure described in the next section.

Issue Report

Enter the following information into the new issue report:

- Time: Add a timestamp for this scenario.
  This could be an instant or a range of time. Search forward through the video footage to find the duration of the scenario if necessary.
- Task: The goal or task is the player trying to complete.

- Description: Create a brief description of the issue.
  For example, if there is a problem with the design, or some difficulty that the player has or could potentially have.

Analyse the issue in more detail using the procedure described in the Issue Analysis section.

Issue Analysis

This stage considers a candidate issue and analyses it using the player action framework tree.

Examine each of the 19 components at the top level of the tree, one by one, and consider whether they're involved in the incident you're currently reviewing. For each of the components that describe a general area involved in this part of the footage, review the heuristics associated to the components. Each heuristic has an associated scenario that describes the criteria for evaluation. Make a judgement about whether each heuristic has been conformed to or violated. Record those heuristics that were violated, and include a brief explanation of why.

Examine each event in the scenario, and indicate whether it's relevant or not to the current issue you're reviewing.

For events that have conditions indicated by sections in parentheses, such as "(DOES/NOT)" or "control" indicate what the condition is:

Most of the events in the scenario can be expressed in positive or negative forms, shown in parentheses, such as "Control (IS/NOT) introduced". For explicit conditions, such as "(DOES/NOT)", indicate the condition that describes the event, e.g., "Player (DOES) understand goal/task".

Create a new issue report with the following information,

- Time.
  Enter either a single point in time, or a range that covers the duration of the scenario.
- Description.
  Write a short summary of the issue.
- Component.
  Enter the name of the component(s) used to evaluate the issue.
- Heuristics.
  For each component list the heuristic(s) that were violated.
- Heuristic violation comment
  Describe how the heuristic(s) were violated.
- Events.
  For each component list the events from the scenario(s) that were violated.

- Event violation comment
  Describe how the event(s) were violated.

C.2 Heuristic Evaluation Procedure

This section introduces the heuristic evaluation method and gives instructions to evaluators for how to evaluate video games using it.

C.2.1 Terminology

"Component"
: Components are the general areas that are considered when analysing usability issues. Each component consists of a short name indicating the topic, example heuristics that

describe relevant design guidelines, and interaction scenarios which represent the typical sequence of interaction.

"Heuristic"
: A heuristic is a simple rule of thumb or guideline that describes common properties of a systems with good usability.

C.2.2 Overview

Using heuristic evaluation involves the following steps:

1. Training.
2. Game play.
3. Issue detection.
4. Issue analysis.

Each of these stages is described in further detail in the following sections.

C.2.3 Training

Evaluators were given a brief reminder about heuristic evaluation. This description was verbally delivered, and is based on Nielsen (1994c).

> Heuristic evaluation is a way of evaluating the usability of a system, such as an application or video game. A heuristic is a simple rule of thumb or guideline that describes common properties of a systems with good usability. The goal of the evaluation is to find usability problems that could affect real users when they use the system in the real world.
>
> During an evaluation the evaluator inspects each element or aspect of the system and compares them to the statements in the heuristics, judging the system for compliance or violation to the heuristics. The evaluator notes all usability problems found and the heuristics that were violated for each of them.

Following this, evaluators reviewed the heuristics and components in the tree hierarchy: Appendix B.2 (Player Action Framework Tree). They were encouraged to ask the facilitator for clarification of any heuristics that were unclear.

C.2.4 Game Play

Play the game and try to think out loud whenever possible. Describe what you're doing in the game, what you think or feel about any aspect that comes to mind. For example, if you like or dislike something, if you're uncertain or feel confident that you know what you have to do next, if you understand or are unclear about anything.

It's important to bear in mind that *you* are not being tested in any way. However you play the game, and whatever you think about it is completely fine and treated as confidential. Please be as honest as direct as possible. Your experience, opinions and comments whether positive or negative are very valuable.

While you're playing consider each element or aspect of the game in relation to the heuristics provided in Appendix C.5 (Player Action Framework - Heuristics). Judge whether the elements in the game conform to or violate any of the heuristics in the list.

Issue Detection

Look for the following cases:

- An actual problem occurs that affects your gameplay.
- A possible problem occurs that does not affect your gameplay, but which you predict could potentially affect other players.

Candidate problems are where play does not proceed in the correct, preferred, expected, or optimal manner. For example, the player makes a mistake, or expresses (verbally or non-verbally) something negative about their game experience.

Whenever you observe either of these cases, pause the game at a convenient point and review all of your previously recorded issues. Has the current issue already been recorded earlier?

- Yes:
  Add the new timestamp (an instant or a duration that the incident occurred over) to the previous report.
  Add additional heuristics if necessary (e.g., count multiple failures) according to the Candidate Analysis procedure following.
- No:
  Create a new report following the procedure described in the next section.

Issue Report

Enter the following information into the new issue report:

- Time: Add a timestamp for this scenario.
  This is the current time according to the clock in the room.
- Task: The goal or task is the player trying to complete.
- Description: Create a brief description of the issue.
  For example, if there is a problem with the design, or some difficulty that the player has or could potentially have.

Analyse the issue in more detail using the procedure described in the Issue Analysis section.

Issue Analysis

This stage considers a candidate issue and analyses it using the player action framework tree.

Examine each of the 19 components at the top level of the tree, one by one, and consider whether they're involved in the incident you're currently reviewing. For each of the components that describe a general area involved in this part of the footage, review the heuristics associated

to the components. Make a judgement about whether each heuristic has been conformed to or violated. Record those heuristics that were violated, and include a brief explanation of why.

Create a new issue report with the following information,

- Time.
  Enter either a single point in time, or a range that covers the duration of the scenario.
- Task: The goal or task is the player trying to complete.
- Description.
  Write a short summary of the issue.
- Components.
  Enter the name of the component(s) used to evaluate the issue.
- Heuristics.
  For each component list the heuristic(s) that were violated.

C.3 Heuristic Evaluation Report Form

Enter the following information to describe the issue:

- Time: When during the recording the event occurred, either an instant or a range of time.
- Issue: A brief description of the problem.
- Task: The goal or task is the player trying to complete.
- Component: The high level component(s) related to the problem area, from Appendix B.2 (Player Action Framework Tree).
- Heuristic: The heuristic(s) related to the Component(s) that describe the general problem.
- Heuristic violation comment: Describing how the heuristic(s) listed were violated.

An example report is shown describing a theoretical problem,

- Timestamp: 15:21:30
- Issue: Player is unable to use the rocket launcher needed to kill the end of level boss. Understands what he needs to do, but doesn't know how to do it. Keeps pressing the wrong buttons. Will probably be able to work it out eventually.
- Task: Defeat the boss.
- Component: Component 4: Usable Controls
- Heuristic: Heuristic 23: "Controls should be intuitive and mapped in a natural way"
- Heuristic violation comment: Player isn't intuitively able to work out the controls.

C.4 Playthrough Evaluation Report Form

- Timestamp: When during the recording the event occurred.
- Issue: Description of the problem.
- Task: The goal or task is the player trying to complete.
- Component: The high level component(s) related to the problem area, from Appendix B.2 (Player Action Framework Tree).
- Heuristic: The heuristic(s) related to the Component(s) that describe the general problem.

- Heuristic violation comment: Describing how the heuristic(s) listed were violated.
- Event code: Any of the events for each heuristic indicated above.
- Event violation comment: Describing how the event(s) listed were violated.

An example report is shown describing a theoretical problem,

- Timestamp: 15:21:30
- Issue: Player is unable to use the rocket launcher needed to kill the end of level boss. Understands what he needs to do, but doesn't know how to do it. Keeps pressing the wrong buttons. Will probably be able to work it out eventually.
- Task: Defeat the boss.
- Component: Component 4: Usable Controls
- Heuristic: Heuristic 23: "Controls should be intuitive and mapped in a natural way"
- Heuristic violation comment: Player isn't intuitively able to work out the controls.
- Event code: Event 23: Player (DOES/NOT) understand how to use control/feature/skill/mechanic
- Event violation comment: Player doesn't understand how to activate the alternate fire on the rocket launcher.
- Event code: Event 7: (UN/necessary/desirable/expected/correct) skill/control/action/mechanic/feature/tactic use (IS/NOT) attempted
- Event violation comment: Player presses the wrong buttons instead, including jump and crouch unnecessarily.

C.5 Player Action Framework - Heuristics

This sections presents the player action framework used in heuristic evaluation, consisting of 19 components and their associated heuristics.

In most cases only a single heuristic and issue were used in the decomposition stage of Chapter 6 (The Playthrough Evaluation Framework). In cases where more than one issue was given the highest rating, all such candidates are were included in the decomposition and are reported here. Further discussion about the process and results of this derivation process are described in detail in Chapter 6 (The Playthrough Evaluation Framework).

1. Component 1: Learning Skills (Controls, Mechanics, Tactics)
    - Heuristic 74: "Player provided with opportunities to practice new skills so as to commit skills to memory"
2. Component 2: Challenge
    - Heuristic 16: "Challenge, strategy, and pace are in balance"
3. Component 13: Balance

- Heuristic 35: "Game play should be balanced so that there is no definite way to win"

4. Component 3: Manual & Tutorial (Help and Documentation)
    - Heuristic 83: "Provide instructions, training, and help"
5. Component 10: Entertainment
    - Heuristic 133: "The players finds the game fun, with no repetitive or boring tasks"
6. Component 6: UI (HUD, Menu)
    - Heuristic 46: "Interfaces should be consistent in control, color, typography, and dialog design"
7. Component 8: Visibility (Visibility of System Status)
    - Heuristic 45: "Indicators are visible"
8. Component 20: Screen Layout
    - Heuristic 92: "Screen layout is efficient, integrated, and visually pleasing"
9. Component 4: Usable Controls
    - Heuristic 23: "Controls should be intuitive and mapped in a natural way"
10. Component 5: Feedback
    - Heuristic 80: "Provide appropriate audio/visual/visceral feedback (music, sound effects, controller vibration)"
11. Component 11: Player in Control
    - Heuristic 76: "Players feel in control"
12. Component 9: Various Styles
    - Heuristic 116: "The game supports different playing styles"
13. Component 15: Unreasonable/Unexpected/Unacceptable Errors (Error Prevention)
    - Heuristic 42: "Help users recognize, diagnose, and recover from errors"
14. Component 18: Audio/Visual Aesthetics
    - Heuristic 119: "The game utilizes visceral, audio and visual content to further the players' immersion in the game"
15. Component 12: Visual Representation Form & Function
    - Heuristic 11: "Art is recognizable to the player and speaks to its function"
16. Component 17: External Consistency (Consistency and Standards)
    - Heuristic 50: "Mechanics should feel natural and have correct weight and momentum"
17. Component 19: Internal Consistency
    - Heuristic 19: "Consistency and Standards"
18. Component 14: Clear Goals
    - Heuristic 104: "The game goals are clear. The game provides clear goals, presents overriding goals early as well as short term goals throughout game play"

19. Component 16: Approachability
    - Heuristic 58: "Player able to master game using skills and tools provided"

Appendix D

User Test Issues

This appendix lists the 88 issues from the user test reported in Chapter 4 (Testing Heuristic Evaluation for Video Games). Each issue lists an issue ID number and description, and in some cases an optional suggestion or further note recorded in the original problem report.

(1) Moderator intervenes to tell the player how to distract.

Player requires help.

(2) Text appears informing of the cloaking feature. It says player will be invisible 'from a distance.' Player immediately walks up to marines and gets shot at.

Is it possible to be more precise about what 'at a distance means'? When asked at the end most players said 'within a few meters', would this text help players to understand how close they can get?

(3) Player tries many times to jump off from part of the level, the graphics suggest that he should be allowed to. Other players have tried this also.

Suggest that the graphics visually enforce that jumping at this point is not possible.

(4) Player says 'It's very difficult to sneak up on people, going slowly etc.' He says his next approach is 'I'll rush in an use the laptop before they can kill me.'

Player comment.

(5) Player comments that 'Stealth attacks are very hard to use, you seem to have to be right behind them.'

Player comment.

(6) Player sees the waypoint on the roof. Mashes buttons, but can't work out how to interact with it.

Help should inform the user what to do at this point.

(7) What is the bar for that sometimes appears beneath the health bar?

Comment. This should be explained to the player.

(8) Player comments that he's not sure what the advantage of stealth kill is, it's not really stealth

(9) Player comments that 'This point seems to have too many aliens in it.'

Difficulty balance seems too steep for this mainstream player (playing in medium difficulty). He dies 13 times at this point.

(10) Player says he's unfamiliar with the triangle (waypoint) symbol. Think it's relevant to the AvP brand.

Tutorial would resolve these issues.

(11) Player comments that he's confused as to where he should be going at the moment. Moderator intervenes to remind him about the focus mode.

The waypoint compass is blank, providing feedback with this could help.

(12) Player comments 'The button controls aren't entirely intuitive', also block on the left bumper can be a little frustrating to use.

Player comment.

(13) Player comments almost immediately that 'There's this thing at the bottom showing me which way to go.'

Players seem to notice the navigation easily on the marine, but not the predator or alien. Tutorial would help this.

(14) Player comments 'I'm trying to figure out what it is that's letting them see me. Is it the sound of my movement or a flashlight?'

Cloaking system needs to be clearly explained in the tutorial, especially how close the predator can get before becoming visible.

(15) Player is not sure if the distract mode worked or not. He said the person did not move.

When activated, the person being distracted should move instantly to provide feedback that the operation has worked.

(16) There are three separate textual messages on the screen simultaneously.

Don't overload the player, suggest to stagger messages if possible.

(17) Humphrey intervenes, shows player where plasmacaster and mines are.

Player intervention.

(18) Player dies in combat. Despite picking up new weapons, he didn't not use them.

Player should have a clear tutorial on weapons as soon as they are obtained.

(19) Player comments that 'I can't interact with this.' Others players have had the same problem. He needs to get the battery first.

If players cannot interact at this stage with an object, provide clear feedback either through text, graphics or audio.

(20) Player comments about the marines that it's difficult to 'get a hold of these people and kill them.'

Perhaps a lock-on system, or other player assistance might help the alien's melee system.

(21) Player is engaged in combat but does not seem to know how to attack.

Tutorial system should introduce combat controls before immersing the player in combat.

(22) Player says as he walks into combat, 'I can't remember how you restore health.'

It seems there are too many controls to remember. The player has not been given time to practice them sufficiently.

(23) Player restarts from the last checkpoint.

Player should not be able to miss objectives.

(24) Player tries to go into stealth mode but fails. He thinks it's because he's used it too much, but has probably just hit the wrong buttons.

Even core players need a way of introducing them to all the controls.

(25) Player checks in-game options after playing for 10 minutes, possibly looking for help or a map. Moderator intervenes to remind him of the objectives screen.

Remind player about objectives screen during early phases of the game.

(26) Objectives screen is telling playing to perform a finishing move. Player comments that he's run out of people to kill. The level is too open for a tutorial, the player spends a lot of time roaming around.

A linear-style level may help to introduce the concepts to the player.

(27) Text appears on the screen when the player is engaged in a battle with an alien.

Not sure players will read this during combat. Perhaps the blocking should be introduced beforehand in a safer context (tutorial).

(28) Player discovers that sometimes others can reject his finishing move. He's not sure if his attack was blocked or if he didn't hold down the right buttons.

Tutorial should explain that sometimes a player's attack can be blocked.

(29) Player comments that 'They should highlight it more when you're about to run out of energy.'

Suggest that the energy part of the HUD catches the players attention when it's running low.

(30) Player comments 'There's something on my HUD and focus mode isn't tell me what it is.' N.B. player is looking at a waypoint marker.

Player is expecting features from focus mode, also he doesn't associate the triangle with a waypoint.

(31) Player comments that 'Sometimes I can't eat the brains of some people that I kill to restore my health, that's a bit annoying.'

Is there a reason for this? Perhaps it could be explained?

(32) Message on screen says 'Harvest the Civilian', player comments that he can't find any civilians. Player spends a lot of time wandering around lost.

The waypoint compass provides no feedback (not present), it should help the user with the objective.

(33) Player comments that 'No matter how far away I am from that marine, he always tracks me with his gun.'

The player is cloaked. The player is not sure how his cloaking behaves (how close, noise etc). This needs explained in the tutorial.

(34) Player comments 'The HUD is still flashing commands at me like 'finish' and 'counter', but all I'm doing is mashing buttons.'

Melee system is not understood (controllable) by the player. Tutorial system should clarify the mechanics.

(35) Player dies. Player goes in 'guns blazing' approach. He does not understand what it means to play as a Predator (stealth approach).

In addition to the character mechanics being explained in the tutorial, perhaps the 'spirit' of the character should be explained also.

(36) Player comments 'The icon at the top of the screen doesn't offer much help, it just seems to point towards the entrance, not the where the third entrance key is.'

Player confusion over what the waypoint compass actually points to. Clarify and be consistent.

(37) Player seems lost, no waypoint indicator at top of screen.

Should waypoint indicator always be present?

(38) Player talks about the canisters again 'I don't understand what they're for. They're obviously important as they've got circles around them.'

(39) The player says he didn't know that he had the plasmacaster or know how to select it. He also didn't realize there was an energy or health meter.

As before, a structured tutorial system could guide the player through these essentials.

(40) Moderator gives tutorial on how to use the alien.

Moderator explains the controls for the character.

(41) 'Counter' message appears, concept has not been introduced.

Does the Marine need all of interrupt / block and counter mechanisms? Possibly suggest to simplify the melee system for the Marine character.

(42) 'Interrupt' message appears on the screen, however this concept has not been introduced.

Does the Marine need all of interrupt / block and counter mechanisms? Possibly suggest to simplify the melee system for the Marine character.

(43) Player picks up a staff, but is not informed as to what it does or how to use it.

Inform player what the staff does and how it can be used (allow them to practice.)

(44) Player comments that the shotgun sounds a bit wimpy at the moment.

Player comment on weapon audio.

(45) Player seems unsure as to what devouring the human heads does.

Provide clear feedback on the purpose (health?)

(46) Waypoint system does not seem to be giving feedback.

Be consistent, player will likely expect the waypoint to always point to next area.

(47) Player throws a mine, but stands on it himself. Don't think he realized what it was.

Perhaps the Predator is immune to their own mines?

(48) Predator seems very easy to kill if the player has the mini-gun.

Observation.

(49) Player comments 'It's hard to see the vents.'

Player comment on signposting / environment.

(50) Player comments 'It seems to be allowing me to do stealth kills in the middle of combat.' Later comments that he can do a stealth kill while not cloaked.

The melee mechanics are not matching the player's expectations. It's possible he will adjust to this over time.

(51) Player comments 'Why can't I leap straight off the edge?'

Visuals of the map do not match what the player expects. If the player cannot jump off here, suggest that graphics convey that.

(52) Player says 'Does that beep mean that I've scanned him?'

Provide clear feedback to the user whenever a scan is successful (audio and / or visual).

(53) Player has difficulty with the platform puzzle. He complains that it's too dark. He dies a few times and can't complete it.

Player comment.

(54) Player doesn't shoot the red tanks hanging from the queen.

Comment. In interview the player said he noticed them, however thought he was killing the queen fine without shooting them.

(55) Red Predator text is very difficult to read, even on an HDTV.

Amount of text should be reduced if possible. Also check readability of text on a standard TV.

(56) Player starts charging his energy, but gets attacked at the same time. He comments 'Maybe some way to cancel that would be useful.'

Allow the player a way to cancel the charging animation.

(57) Player comments that 'Every time I do a finishing move, he just pulls away every time. Not sure I want to use a finishing move.'

Player comment on melee system.

(58) Player comments 'Ahh that's what the they're wielding'. He's just worked out what the UI is trying to tell him.

Players could benefit from a clear tutorial.

(59) Player comments 'How do I attack him.'

Player either didn't read the instructions or cannot remember them. Suggest to let the player practice using the controls immediately after they're introduced.

(60) Player comments that 'It wants me to block at times, but it seems rather pointless.'

Comment, does the block mechanic make sense for the alien character?

(61) Player seems lost, he comments 'I guess I need to go up, that arrow keeps pointing up.'

Not sure the player understands the waypoint arrow. This should be covered in the tutorial.

(62) Player comments 'There are arrows here but I'm not sure what they are.'

Game features should be clearly explained to the player.

(63) Player comments that getting in to the vent was difficult. A few seconds later he comments again that he can't get down a hole.

Player comment on basic movement. Players should be able to just press A to go through a vent / hole if they're up close.

(64) Player comments that the hacking scene takes a bit long, seems drawn out.

Player comment.

(65) Player comments 'OK I've forgotten which button it said I should press to go on to the ceiling or whatever.'

Whenever a new mechanic is introduced, allow the player to practice it immediately.

(66) Player comments 'I'm not quite sure how to overload this power node.'

Objectives may need to provide clearer statements, or tutorial should introduce how to overload power nodes.

(67) Moderator intervenes to explain the play mechanics. Even when given guidance, the player still finds difficulty in using all the controls of the predator (scanning, zooming, cloaking, distracting, vision modes and light / heavy attack.)

This character is the most complicated, the tutorial should be well paced (i.e. slow but balanced) and give the player ample opportunity to learn all the features.

(68) Player reads the objectives, 'It says I should harvest but I don't know what that is.'

Player comment.

(69) Player comments that he can 'See ammo but can't pick it up which is annoying.'

Player comment.

(70) Player activates a switch then asks 'Ok, what did that do, turn off electricity somewhere?'

Provide feedback (text or video cut) to users as to what switches etc do.

(71) Player goes straight into 3 marines, hasn't attempted scanning yet.

Player has not grasped how to best use the predator character. Tutorial should guide through this process.

(72) Player says stealth kill didn't work.

Predator's mechanics are not intuitive, seems players will need a thorough tutorial.

(73) Player says 'It doesn't repeat the tutorial over again, if you don't get it first time...'

Tutorial system should 'check' if the player has practiced the features, if not possibly remind the player again.

(74) Player comments that 'The marine moves slower than other characters in games such as Far Cry, Halo or CoD.'

Player comment.

(75) Player comments that the speed of the scope zoom is a little bit slow, speed it up.

Player comment on weapons.

(76) Player comments 'The camera angles on the alien can be very, very annoying when walking around on walls.'

Player comment on character control.

(77) Player comments that 'Fast attack seems to take him down a lot faster (than heavy)'.

Suggest to make heavy attack more powerful, a least a perceivable difference to the player.

(78) Moderator intervenes to remind player about cloaking. Player comments that he was playing 'in the style of the alien.'

Player needs help. Evidence also that the player should be thinking of how to play in the 'style' of each character.

(79) Player comments 'OK I've just picked something up', however he just walked over the waypoint marker.

Navigation and waypoint needs to be explained in the tutorial.

(80) Player comments that he would like a way of seeing the aliens in the dark. He goes into the settings and turns up the brightness.

Should the player have the alien vision mode at this stage?

(81) Player comments 'I wonder where my life is.'

Player comment on UI.

(82) Player comments 'It said pressed LB for something, but it was pretty quick.'

Leave message on-screen for longer, especially during the intro levels.

(83) Player activates a switch then comments 'Ok what did that do' after looking around for a while.

Provide clear feedback to the user when activating switches, what is the result of their actions?

(84) Player comments 'I'm not sure why I'm dying when I'm hitting people. Maybe I'm not blocking enough.'

An understanding of the combat system should be introduced.

(85) Moderator intervenes to explain the radar system.

Player assist.

(86) Player comments 'I picked up a gun earlier, but I don't know how I can use it.'

Allow players the opportunity to practice with weapons right away after they pick them up. Tutorial should guide them in usage.

(87) Player seems to be focus jumping onto enemies, he seems confused over the character controls.

Player hasn't grasped basic understanding of character control / melee. Tutorial should cover these essentials.

(88) Player comments that 'There are some aliens who don't show up in thermal imaging', don't think he understands what they are.

Player comment.

Appendix E

146 Heuristics

The follow lists all of the 146 heuristics used in Chapter 4 (Testing Heuristic Evaluation for Video Games),

(1) A good game should be easy to learn and hard to master (Nolan Bushnell).

(2) A player should always be able to identify their score/status in the game.

(3) Actions and skills learned were important for playing the game not just for a single event in the game.

(4) Aesthetic and minimalist design.

(5) AI is balanced with the players' play.

(6) All levels of players are able to play and get involved quickly and easily with tutorials, and/or progressive or adjustable difficulty levels.

(7) Allow players to build content.

(8) Allow users to customize video and audio settings, difficulty and game speed.

(9) Allow users to skip non-playable and frequently repeated content.

(10) Any fatigue or boredom was minimized by varying activities and pacing during the game play.

(11) Art is recognizable to the player and speaks to its function.

(12) Art should speak to its function.

(13) Artificial intelligence should be reasonable yet unpredictable.

(14) Audio-visual representation supports the game.

(15) Build as though the world is going on whether your character is there or not.

(16) Challenge, strategy, and pace are in balance.

(17) Challenges are positive game experiences, rather than negative experiences, resulting in wanting to play more, rather than quitting.

(18) Changes the player make in the game world are persistent and noticeable if they back-track to where they have been before.

(19) Consistency and Standards.

(20) Consistency shortens the learning curve by following the trends set by the gaming industry to meet users' expectations. If no industry standard exists, perform usability / playability research to ascertain the best mapping for the majority of intended players.

(21) Control keys are consistent and follow standard conventions.

(22) Controls should be customizable and default to industry standard settings.

(23) Controls should be intuitive and mapped in a natural way.

(24) Create a great storyline.

(25) Design for multiple paths through the game.

(26) Device UI and game UI are used for their own purposes.

(27) Do not expect the user to read a manual.

(28) Easy to learn, harder to master.

(29) Error Prevention.

(30) Feedback should be given immediately to display user control.

(31) Flexibility and efficiency of use.

(32) Follow the trends set by the gaming community to shorten the learning curve.

(33) For PC games, consider hiding the main computer interface during game play.

(34) Game controls are convenient and flexible.

(35) Game play should be balanced so that there is no definite way to win.

(36) Game provides feedback and reacts in a consistent, immediate, challenging and exciting way to the players' actions.

(37) Game story encourages immersion (If game has story component).

(38) Gameplay is long and enduring and keeps the players' interest.

(39) Games similar to others in same genre allowing new skills to be built on previous knowledge.

(40) Get the player involved quickly and easily.

(41) Help and documentation.

(42) Help users recognize, diagnose, and recover from errors.

(43) If the game cannot be modeless, it should feel modeless to the player.

(44) Include a lot of interactive props for the player to interact with.

(45) Indicators are visible.

(46) Interfaces should be consistent in control, color, typography, and dialog design.

(47) Learning curve is too steep; requires too much micromanagement; command sequences are complex, lengthy, and awkward, making the game difficult to play.

(48) Make the game replayable.

(49) Match between the system and the real world.

(50) Mechanics should feel natural and have correct weight and momentum.

(51) Minimize control options.

(52) Minimize the menu layers of an interface.

(53) Navigation is consistent, logical, and minimalist.

(54) One reward of playing should be the acquisition of skill.

(55) Pace the game to apply pressure to, but not frustrate the player.

(56) Play should be fair.

(57) Player able to demonstrate and practice new actions without severe consequences. Player knows what actions to take.

(58) Player able to master game using skills and tools provided.

(59) Player able to succeed at game's goals and found their expectations fulfilled.

(60) Player able to succeed at playing game after training period, i.e., first level or tutorial.

(61) Player able to use preferred style.

(62) Player affects the game world.

(63) Player does not need to access the tutorial in order to play.

(64) Player does not need to read the manual or documentation to play.

(65) Player error is avoided.

(66) Player feels rewards and punishments for game play action were appropriate.

(67) Player given increased capabilities/tools to use.

(68) Player given opportunity to model correct behavior and skills.

(69) Player has access to answers re: the game whenever needed and when first coming across new material.

(70) Player identifies with character.

(71) Player interruption is supported, so that players can easily turn the game on and off and be able to save the games in different states.

(72) Player is given controls that are basic enough to learn quickly, yet expandable for advanced options for advanced players.

(73) Player provided with help to meet goals of game.

(74) Player provided with opportunities to practice new skills so as to commit skills to memory.

(75) Player was entertained and enjoyed playing the game.

(76) Players feel in control.

(77) Players should be able to save games in different states.

(78) Players should be given context sensitive help while playing so that they are not stuck and need to rely on a manual for help.

(79) Provide an interesting and absorbing tutorial.

(80) Provide appropriate audio/visual/visceral feedback (music, sound effects, controller vibration).

(81) Provide consistent responses to the users actions.

(82) Provide controls that are easy to manage, and that have an appropriate level of sensitivity and responsiveness.

(83) Provide instructions, training, and help.

(84) Provide intuitive and customizable input mappings.

(85) Provide means for error prevention and recovery through the use of warning messages.

(86) Provide predictable and reasonable behavior for computer controlled units.

(87) Provide unobstructed views that are appropriate for the users current actions.

(88) Provide users with information on game status.

(89) Provide visual representations that are easy to interpret and that minimize the need for micromanagement.

(90) Recognition rather than recall.

(91) Screen layout is efficient and visually pleasing.

(92) Screen layout is efficient, integrated, and visually pleasing.

(93) Should use visual and audio effects to arouse interest.

(94) Status score Indicators are seamless, obvious, available and do not interfere with game play.

(95) Teach skills early that you expect the players to use later.

(96) The AI is tough enough that the players have to try different tactics against it.

(97) The first ten minutes of play and player actions are painfully obvious and should result in immediate and positive feedback for all types of players.

(98) The first-time experience is encouraging.

(99) The game contains help.

(100) The game does not put an unnecessary burden on the player.

(101) The game does not stagnate.

(102) The game gives feedback on the player's actions.

(103) The game gives rewards that immerse the player more deeply in the game by increasing their capabilities, capacity or for example, expanding their ability to customize.

(104) The game goals are clear. The game provides clear goals, presents overriding goals early as well as short term goals throughout game play.

(105) The game had different AI settings so that it was challenging to all levels of players, whether novice or expert players.

(106) The game is balanced with multiple ways to win.

(107) The game is consistent.

(108) The game is paced to apply pressure without frustrating the players. The difficulty level varies so the players experience greater challenges as they develop mastery.

(109) The game offers something different in terms of attracting and retaining the players' interest.

(110) The game provides clear goals or supports player-created goals.

(111) The game should give hints, but not too many.

(112) The game should give rewards.

(113) The game should have an unexpected outcome.

(114) The game story supports the gameplay and is meaningful.

(115) The game supports a variety of game styles.

(116) The game supports different playing styles.

(117) The game uses humor well.

(118) The game uses orthogonal unit differentiation.

(119) The game utilizes visceral, audio and visual content to further the players' immersion in the game.

(120) The game world reacts to the player and remembers their passage through it.

(121) The interface should be as non-intrusive as possible.

(122) The player cannot make irreversible errors.

(123) The player does not have to memorize things unnecessarily.

(124) The player does not lose any hard-won possessions.

(125) The player experiences the user interface as consistent (in controller, color, typographic, dialogue and user interface design).

(126) The player is in control.

(127) The player sees the progress in the game and can compare the results.

(128) The player understands the terminology.

(129) The player's have a sense of control and influence onto the game world.

(130) The players are rewarded and rewards are meaningful.

(131) The players can express themselves.

(132) The players experience the user interface/HUD as a part of the game.

(133) The players finds the game fun, with no repetitive or boring tasks.

(134) The players should not experience being penalized repetitively for the same failure.

(135) The players should not lose any hard won possessions.

(136) The skills needed to attain goals are taught early enough to play or use later, or right before the new skill is needed.

(137) There are no repetitive or boring tasks.

(138) There is an emotional connection between the player and the game world as well as with their 'avatar.'.

(139) There must not be any single optimal winning strategy.

(140) There should be a clear overriding goal of the game presented early.

(141) There should be multiple goals on each level.

(142) There should be variable difficulty level.

(143) Upon turning on the game, the player has enough information to begin play.

(144) Use sound to provide meaningful feedback.

(145) User Control and Freedom.

(146) Visibility of System Status.

Bibliography

Andre, Terence S. (Apr. 2000). "Determining the Effectiveness of the Usability Problem Inspector: A Theory-Based Model and Tool for Finding Usability Problems". Ph.D. thesis. Blacksburg, Virginia: Industrial and Systems Engineering (cited on pp. 67, 116, 228).

Andre, Terence S., Steven M. Belz, Faith A. McCreary, and H. Rex Hartson (July 2000). "Testing a Framework for Reliable Classification of Usability Problems". In: *Human Factors and Ergonomics Society Annual Meeting Proceedings* 44, 573–576(4). URL: http://www.ingentaconnect.com/content/hfes/hfproc/2000/00000044/00000037/art00007 (cited on pp. 116, 228).

Andre, Terence S., H. Rex Hartson, Steven M. Belz, and Faith A. McCreary (2001). "The user action framework: a reliable foundation for usability engineering support tools". In: *International Journal of Human-Computer Studies* 54.1, pp. 107 –136. ISSN: 1071-5819. DOI: DOI: 10.1006/ijhc.2000.0441. URL: http://www.sciencedirect.com/science/article/B6WGR-458NDY6-V/2/fcbb81c1fc7f793aeb98da98f91ff3ef (cited on pp. 63–64, 116–117, 228).

Andre, Terence S., H. Rex Hartson, and Robert C. Williges (2003). "Determining the Effectiveness of the Usability Problem Inspector: A Theory-Based Model and Tool for Finding Usability Problems". In: *Human Factors* 45.3, pp. 455–482 (cited on pp. 116, 228).

Baauw, Ester, Mathilde Bekker, and Wolmet Barendregt (2005). "A Structured Expert Evaluation Method for the Evaluation of Children's Computer Games". In: *Human-Computer Interaction - INTERACT 2005*. Ed. by Maria Costabile and Fabio Paternò. Vol. 3585. Lecture Notes in Computer Science. Berlin / Heidelberg: Springer, pp. 457–469. URL: http://dx.doi.org/10.1007/11555261_38 (cited on pp. 3, 77, 116).

Barendregt, W (Jan. 2006). "Evaluating fun and usability in computer games with children". Ph.D. thesis. URL: http://alexandria.tue.nl/extra2/200513731.pdf (cited on pp. 58, 95, 128).

Barendregt, W. and M. Bekker (2006). "Developing a coding scheme for detecting usability and fun problems in computer games for young children". In: *Behavior Research Methods* 38 (3), pp. 382–389. ISSN: 1554-351X. URL: http://dx.doi.org/10.3758/BF03192791 (cited on pp. 59, 128).

Barendregt, Wolmet, Mathilde M. Bekker, Don Bouwhuis, and Esther Baauw (2007). "Predicting effectiveness of children participants in user testing based on personality characteristics". In: *Behaviour & Information Technology* 26, pp. 133–147. DOI: 10.1080/01449290500330372 (cited on p. 128).

Barendregt, Wolmet, Mathilde M. Bekker, and Mathilde Speerstra (2003). "Empirical evaluation of usability and fun in computer games for children". In: *Proc. Human-Computer Interaction - INTERACT '03*. Ed. by M. Rauterberg et al. Vol. 3. IOS Press, pp. 705–708. URL: http://www.idemployee.id.tue.nl/g.w.m.rauterberg/conferences/INTERACT2003/INTERACT2003-p705.pdf (cited on pp. 22, 68, 77).

Barendregt, Wolmet, M M Bekker, D G Bouwhuis, and E Baauw (2006). "Identifying usability and fun problems in a computer game during first use and after some practice". In: *Int. J. Hum.-Comput. Stud.* 64.9, pp. 830–846. URL: http://dx.doi.org/10.1016/j.ijhcs.2006.03.004 (cited on pp. 36, 59, 128).

Bargas-Avila, Javier A. and Kasper Hornbæk (2011). "Old wine in new bottles or novel challenges: a critical analysis of empirical studies of user experience". In: *Proceedings of the 2011 annual conference on Human factors in computing systems*. CHI '11. Vancouver, BC, Canada: ACM, pp. 2689–2698. ISBN: 978-1-4503-0228-9. DOI: http://doi.acm.org/10.1145/1978942.1979336. URL: http://doi.acm.org/10.1145/1978942.1979336 (cited on p. 64).

Barr, Pippin (2008). "Video Game Values Play as Human-Computer Interaction". Ph.D. thesis. Victoria University of Wellington. URL: http://www.pippinbarr.com/academic/Pippin_Barr_PhD_Thesis.pdf (cited on p. 22).

— (Mar. 2010a). *Cognitive Playthrough Notes from Lab Session*. URL: https://blog.itu.dk/MAHS-F2010/2010/03/08/cognitive-playthrough-notes-from-lab-session/ (cited on p. 223).

— (Mar. 2010b). *The Cognitive Playthrough*. URL: https://blog.itu.dk/MAHS-F2010/2010/03/08/the-cognitive-playthrough/ (cited on pp. 22, 77, 223).

Barr, Pippin, Rilla Khaled, James Noble, and Robert Biddle (2006). "Playing the interface: a case study of Grand Theft Auto: San Andreas". In: *OZCHI '06: Proceedings of the 18th Australia conference on Computer-Human Interaction*. Sydney, Australia: ACM, pp. 317–320. ISBN: 1-59593-545-2. DOI: http://doi.acm.org/10.1145/1228175.1228233 (cited on p. 22).

Barr, Pippin, James Noble, and Robert Biddle (2007). "Video game values: Human-computer interaction and games". In: *Interact. Comput.* 19.2, pp. 180–195. ISSN: 0953-5438. DOI: http://dx.doi.org/10.1016/j.intcom.2006.08.008 (cited on p. 22).

Bartle, Richard (1996). "Hearts, Clubs, Diamonds, Spades: Players who suit MUDs". In: URL: http://www.mud.co.uk/richard/hcds.htm (cited on p. 23).

Bastien, J. M. Christian and Dominique L. Scapin (Apr. 1995). "Evaluating a user interface with ergonomic criteria". In: *Int. J. Hum.-Comput. Interact.* 7.2, pp. 105–121. ISSN: 1044-7318. DOI: 10.1080/10447319509526114. URL: http://dx.doi.org/10.1080/10447319509526114 (cited on p. 57).

Bateman, Chris and Richard Boon (2005). *21st Century Game Design (Game Development Series)*. Rockland, MA, USA: Charles River Media, Inc. ISBN: 1584504293 (cited on p. 87).

Bernhaupt, R, M Eckschlager, and M Tscheligi (Jan. 2007). "Methods for evaluating games: how to measure usability and user experience in games?" In: *Proceedings of the international conference on Advances in Computer Entertainment*. URL: http://portal.acm.org/citation.cfm?id=1255142 (cited on p. 22).

Bernhaupt, Regina, Wijand Ijsselsteijn, Florian 'Floyd' Mueller, Manfred Tscheligi, and Dennis Wixon (2008). "Evaluating user experiences in games". In: *CHI '08: Proceeding of the twenty-sixth annual CHI conference extended abstracts on Human factors in computing systems*. Florence, Italy: ACM, pp. 3905–3908. ISBN: 978-1-60558-012-X. DOI: http://doi.acm.org/10.1145/1358628.1358953 (cited on p. 22).

Bolton, M. L. and E. J. Bass (2010). "Using Task Analytic Models and Phenotypes of Erroneous Human Behavior to Discover System Failures Using Model Checking". In: *Proceedings of the 54th Annual Meeting of the Human Factors and Ergonomics Society*. Santa Monica: Human Factors and Ergonomics Society, pp. 992–996. URL: http://www.ingentaconnect.com/content/hfes/hfproc/2010/00000054/00000013/art00005 (cited on p. 33).

Brown, Emily and Paul Cairns (2004). "A grounded investigation of game immersion". In: *CHI '04: CHI '04 extended abstracts on Human factors in computing systems*. Vienna, Austria: ACM, pp. 1297–1300. ISBN: 1-58113-703-6. DOI: http://doi.acm.org/10.1145/985921.986048 (cited on pp. 1, 13).

Budge, Bill (1983). *Pinball Construction Set*. BudgeCo (cited on p. 45).

Caillois, Roger (1961). *Man, Play and Games*. Ed. by (trans. Meyer Barash). University of Illinois Press (cited on p. 4).

Capra, Miranda Galadriel (Oct. 2001). "An Exploration of End-User Critical Incident Classification". MA thesis. Industrial and Systems Engineering (cited on pp. 116, 228).

Capra, Miranda Galadriel (Mar. 2006). "Usability Problem Description and the Evaluator Effect in Usability Testing". Ph.D. thesis. Industrial and Systems Engineering (cited on pp. 32, 37, 59).

Card, Stuart K., Thomas P. Moran, and Allen Newell (1980). "The keystroke-level model for user performance time with interactive systems". In: *Commun. ACM* 23.7, pp. 396–410. ISSN: 0001-0782. DOI: http://doi.acm.org/10.1145/358886.358895 (cited on pp. 46, 225).

Card, Stuart K., Allen Newell, and Thomas P. Moran (1983). *The Psychology of Human-Computer Interaction*. Hillsdale, NJ, USA: L. Erlbaum Associates Inc. ISBN: 0898592437 (cited on p. 224).

Carroll, John M., ed. (1995). *Scenario-based design: envisioning work and technology in system development*. New York, NY, USA: John Wiley & Sons, Inc. ISBN: 0-471-07659-7 (cited on pp. 118, 227).

Carroll, John M. and John M. Thomas (1988). "FUN". In: *SIGCHI Bull.* 19.3, pp. 21–24. ISSN: 0736-6906. DOI: http://doi.acm.org/10.1145/49108.1045604 (cited on p. 23).

Catanzaro, Christopher D. (Apr. 2005). "Vizability: Visualizing Usability Evaluation Data Based on the User Action Framework". MA thesis. Blacksburg, Virginia: Virginia Polytechnic Institute and State University (cited on pp. 116, 228).

Cattell, R.B. (1966). "The meaning and strategic use of factor analysis". In: Rand McNally psychology series. Rand McNally (cited on p. 89).

Chattratichart, Jarinee and Gitte Lindgaard (2008). "A comparative evaluation of heuristic-based usability inspection methods". In: *CHI '08 extended abstracts on Human factors in computing systems*. CHI '08. Florence, Italy: ACM, pp. 2213–2220. ISBN: 978-1-60558-012-X. DOI: http://doi.acm.org/10.1145/1358628.1358654. URL: http://doi.acm.org/10.1145/1358628.1358654 (cited on p. 37).

Cockton, Gilbert (2012). "Usability Evaluation". In: *Encyclopedia of Human-Computer Interaction*. Ed. by Mads Soegaard and Rikke Friis Dam. Aarhus, Denmark: The Interaction Design Foundation. URL: http://www.interaction-design.org/encyclopedia/usability_evaluation.html (cited on pp. 19, 23–24, 26, 29, 31, 41, 45, 76, 153, 224, 226).

Cockton, Gilbert, Darryn Lavery, and Alan Woolrych (2003). "Inspection-based evaluations". In: ed. by Julie A. Jacko and Andrew Sears. Hillsdale, NJ, USA: L. Erlbaum Associates Inc. Chap. V.C (57), pp. 1118–1138. ISBN: 0-8058-3838-4 (cited on p. 60).

Cockton, Gilbert and Alan Woolrych (2001). "Understanding inspection methods: Lessons from an assessment of heuristic evaluation". In: *Joint Proceedings of HCI 2001 and IHM 2001: People and Computers XV*, pp. 171–191. URL: http://osiris.sunderland.ac.uk/~cs0awo/hci\%202001\%20full.pdf (cited on pp. 60, 66–67, 96–97).

— (2009). "Comparing Usability Evaluation Methods: Strategies and Implementation (Final Report of COST294-MAUSE Working Group 2)". In: *COST294-MAUSE Closing Conference Proceedings*. Ed. by E. Law, D. Scapin, G. Cockton, M. Springett, C. Stary, and M. Winckler. COST, pp. 18–82 (cited on p. 20).

Cockton, Gilbert, Alan Woolrych, Lynne Hall, and Mark Hindmarch (Sept. 2003). "Changing Analysts' Tunes: The Surprising Impact of a New Instrument for Usability Inspection Method Assessment". In: *Proc. HCI - People and Computers XVII: Designing for Society*. Ed. by P. Palanque, P. Johnson, and E. O'Neill. Springer-Verlag, pp. 145–162 (cited on pp. 6, 34, 49, 102, 110, 224).

Cockton, Gilbert, Alan Woolrych, and Mark Hindmarch (2004). "Reconditioned merchandise: extended structured report formats in usability inspection". In: *CHI '04: CHI '04 extended abstracts on Human factors in computing systems*. Vienna, Austria: ACM, pp. 1433–1436. ISBN: 1-58113-703-6. DOI: http://doi.acm.org/10.1145/985921.986083 (cited on pp. 49, 66, 87, 97, 110).

Cockton, G and D Lavery (1999). "A framework for usability problem extraction". In: *Proc. HCI Interact*. Ed. by MA Sasse and C Johnson. Amsterdam, Netherlands, pp. 344–352. ISBN: 0-9673355-0-7 (cited on pp. 3, 13, 38, 50, 63, 70, 96).

Cohen, Jacob (1960). "A Coefficient of Agreement for Nominal Scales". In: *Educational and Psychological Measurement* 20.1, pp. 37–46. DOI: 10.1177/001316446002000104. eprint: http://epm.sagepub.com/content/20/1/37.full.pdf+html. URL: http://epm.sagepub.com/content/20/1/37.short (cited on p. 223).

Cole, Frank L. (1988). "Content Analysis: Process and Application". In: *Clinical Nurse Specialist* 2.1. URL: http://journals.lww.com/cns-journal/Fulltext/1988/00210/Content_Analysis__Process_and_Application.25.aspx (cited on p. 106).

Connell, Iain W. and N. V. Hammond (Aug. 1999). "Comparing usability evaluation principles with heuristics: problem instances vs. problem types". In: *Proceedings of INTERACT'99 - Human Computer Interaction*, pp. 621–629 (cited on pp. 67, 150).

Desurvire, H, M Caplan, and J Toth (Jan. 2004). "Using heuristics to evaluate the playability of games". In: *Conference on Human Factors in Computing Systems*. URL: http://portal.acm.org/citation.cfm?id=986102 (cited on pp. 25, 66, 68, 71, 85).

Desurvire, Heather and Bernard Chen (2008). *48 Differences Between Good and Bad Video Games: Game Playability Principles (PLAY) For Designing Highly Ranked Video Games*. URL: http://www.behavioristics.com/downloads/PLAYPrinciples-HDesurvire.pdf (cited on p. 85).

Desurvire, Heather and Charlotte Wiberg (2008). "Master of the game: assessing approachability in future game design". In: *Proc. CHI extended abstracts*. Florence, Italy: ACM, pp. 3177–3182. ISBN: 978-1-60558-012-X. DOI: http://doi.acm.org/10.1145/1358628.1358827 (cited on p. 85).

— (2009). "Game Usability Heuristics (PLAY) for Evaluating and Designing Better Games: The Next Iteration". In: *Proc. OCSC (Part of HCI International)*. San Diego, CA: Springer-Verlag, pp. 557–566. ISBN: 978-3-642-02773-4. DOI: http://dx.doi.org/10.1007/978-3-642-02774-1_60 (cited on pp. 10, 24, 85–86, 103, 119, 121).

— (2010). "User Experience Design for Inexperienced Gamers: GAP - Game Approachability Guidelines". In: *Evaluating User Experience in Games - Concepts and Methods*. Ed. by Regina Bernhaupt. Springer-Verlag. Chap. 8 (cited on pp. 68, 71, 85–86, 119).

Doubleday, Ann, Michele Ryan, Mark Springett, and Alistair Sutcliffe (1997). "A comparison of usability techniques for evaluating design". In: *Proceedings of the 2nd conference on Designing interactive systems: processes, practices, methods, and techniques*. DIS '97. Amsterdam, The Netherlands: ACM, pp. 101–110. ISBN: 0-89791-863-0. DOI: http://doi.acm.org/10.1145/263552.263583. URL: http://doi.acm.org/10.1145/263552.263583 (cited on pp. 42, 70, 73–74, 103, 114).

Fabricatore, Carlo (1999). "Playability In Action Videgames: A Theoretical Design Reference". Ph.D. thesis. Pontificia Universidad Catolica De Chile Escuela De Ingenieria, Santiago de Chile: Departamento de Ciencias de la Computación (cited on pp. 29, 92).

Fabricatore, Carlo, Miguel Nussbaum, and Ricardo Rosas (Feb. 2002). "Playability in Action Videogames: A Qualitative Design Model". In: *Human-Comp. Interaction* 17.4, pp. 311–368. DOI: 10.1207/S15327051HCI1704_1 (cited on pp. 29, 87).

Falstein, Noah and Hal Barwood (Mar. 2006). *The 400 Project*. URL: http://www.theinspiracy.com/400_project.htm (cited on p. 87).

Febretti, Alessandro and Franca Garzotto (2009). "Usability, playability, and long-term engagement in computer games". In: *CHI EA '09: Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*. Boston, MA, USA: ACM, pp. 4063–4068. ISBN: 978-1-60558-247-4. DOI: http://doi.acm.org/10.1145/1520340.1520618 (cited on p. 25).

Federoff, M (Jan. 2002). "Heuristics and usability guidelines for the creation and evaluation of fun in video games". MA thesis. Department of Telecommunications, Indiana University. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.89.8294&rep=rep1&type=pdf (cited on pp. 24, 85–86, 119).

Fleiss, Joseph L. (1971). "Measuring nominal scale agreement among many raters". In: *Psychological Bulletin* 76.5, pp. 378–382. URL: http://search.proquest.com/docview/614289059?accountid=14182 (cited on p. 59).

Følstad, Asbjørn, Effie Law, and Kasper Hornbæk (2012). "Analysis in Practical Usability Evaluation: A Survey Study". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '12. Austin, Texas, USA: ACM, pp. 2127–2136. ISBN: 978-1-4503-1015-4. DOI: 10.1145/2207676.2208365. URL: http://doi.acm.org/10.1145/2207676.2208365 (cited on pp. 151–152).

Frasca, Gonzalo (July 2001). *What is ludology? A provisory definition*. Tech. rep. Ludology.org (cited on p. 226).

Freelon, Deen (Sept. 2008). *ReCal for Ordinal, Interval, and Ratio Data (OIR)*. URL: http://dfreelon.org/utils/recalfront/recal-oir/ (cited on p. 87).

Frøkjær, Erik and Kasper Hornbæk (Jan. 2008). "Metaphors of human thinking for usability inspection and design". In: *ACM Trans. Comput.-Hum. Interact.* 14.4, 20:1–20:33. ISSN: 1073-0516. DOI: 10.1145/1314683.1314688. URL: http://doi.acm.org/10.1145/1314683.1314688 (cited on p. 59).

Gerhardt-Powals, Jill (Apr. 1996). "Cognitive engineering principles for enhancing human-computer performance". In: *Int. J. Hum.-Comput. Interact.* 8.2, pp. 189–211. ISSN: 1044-7318. DOI: 10.1080/10447319609526147. URL: http://dx.doi.org/10.1080/10447319609526147 (cited on p. 72).

Giddings, Seth (2006). "Walkthrough: videogames and technocultural form". Ph.D. thesis. The University of the West of England, Bristol: School of Cultural Studies (cited on p. 24).

Gram, C. and G. Cockton (1996). *Design Principles for Interactive Software*. Ifip International Federation for Information Processing. Chapman & Hall. ISBN: 9780412724701. URL: http://books.google.co.uk/books?id=A7a-RVFhvFkC (cited on p. 40).

Gray, Wayne D. and Marilyn C. Salzman (1998). "Damaged merchandise? a review of experiments that compare usability evaluation methods". In: *Hum.-Comput. Interact.* 13.3, pp. 203–261. ISSN: 0737-0024. DOI: http://dx.doi.org/10.1207/s15327051hci1303_2 (cited on pp. 19, 41–42, 47–48, 51–53, 61, 228).

Grudin, Jonathan (Oct. 1989). "The case against user interface consistency". In: *Commun. ACM* 32 (10), pp. 1164–1173. ISSN: 0001-0782. DOI: http://doi.acm.org/10.1145/67933.67934. URL: http://doi.acm.org/10.1145/67933.67934 (cited on pp. 73, 103).

Ham, Dong-Han (2008). "A New Framework for Characterizing and Categorizing Usability Problems". In: *EKC2008 Proceedings of the EU-Korea Conference on Science and Technology*. Ed. by Seung-Deog Yoo. Vol. 124. Springer Proceedings in Physics. Springer Berlin Heidelberg, pp. 345–353. ISBN: 978-3-540-85190-5. URL: http://dx.doi.org/10.1007/978-3-540-85190-5_36 (cited on p. 19).

Hartson, H. Rex, Terence S. Andre, and Robert C. Williges (2001). "Criteria For Evaluating Usability Evaluation Methods". In: *International Journal of Human-Computer Interaction* 13.4, pp. 373–410. URL: http://www.informaworld.com/10.1207/S15327590IJHC1304_03 (cited on pp. 40, 60).

Hartson, H. Rex, Terence S. Andre, Robert C. Williges, and Linda van Rens (1999). "The User Action Framework: A Theory-Based Foundation for Inspection and Classification of Usability Problems". In: *Proceedings of HCI International (the 8th International Conference on Human-Computer Interaction) on Human-Computer Interaction: Ergonomics and User Interfaces-Volume I*. Hillsdale, NJ, USA: L. Erlbaum Associates Inc., pp. 1058–1062. ISBN: 0-8058-3391-9 (cited on pp. 116, 228).

Hartson, Rex (Sept. 2003). "Cognitive, physical, sensory, and functional affordances in interaction design". In: *Behaviour and Information Technology* 22, 315–338(24) (cited on pp. 32, 223).

Hassenzahl, Marc, Markus Schöbel, and Tibor Trautmann (2008). "How motivational orientation influences the evaluation and choice of hedonic and pragmatic interactive products: The role of regulatory focus". In: *Interact. Comput.* 20.4-5, pp. 473–479. ISSN: 0953-5438. DOI: http://dx.doi.org/10.1016/j.intcom.2008.05.001 (cited on p. 23).

Hassenzahl, Marc and Noam Tractinsky (2006). "User experience - a research agenda". In: *Behaviour & Information Technology* 25.2, pp. 91–97. DOI: 10.1080/01449290500330331. eprint: http://www.tandfonline.com/doi/pdf/10.1080/01449290500330331. URL: http://www.tandfonline.com/doi/abs/10.1080/01449290500330331 (cited on p. 228).

Hayes, Andrew F. and Klaus Krippendorff (2007). "Answering the call for a standard reliability measure for coding data". In: *Communication Methods and Measures* 1.1, pp. 77–89 (cited on pp. 87, 225).

Hertzum, Morten and Niels Ebbe Jacobsen (1999). "The evaluator effect during first-time use of the cognitive walkthrough technique". In: *in Proc. 8th Intl. Conf. Human-Computer Interaction, (HCI International '99)*. Erlbaum, pp. 1063–1067 (cited on p. 150).

— (2001). "The evaluator effect: a chilling fact about usability evaluation methods". In: *Int. Journal of Human-Computer Interaction* 13.4, pp. 421–443 (cited on pp. 19, 47–48, 58–60, 128, 150, 223–224, 230).

— (2003). "The Evaluator Effect: A Chilling Fact About Usability Evaluation Methods". In: *International Journal of Human-Computer Interaction* 15.1, pp. 183–204. URL: http://www.informaworld.com/10.1207/S15327590IJHC1501_14 (cited on pp. 59, 128).

Herzberg, F. (1973). *Work and the nature of man*. Mentor book,m. New American Library. URL: http://books.google.co.uk/books?id=ClFhAAAAIAAJ (cited on p. 23).

Hix, D. and H.R. Hartson (1993). "Developing user interfaces: ensuring usability through product & process". In: Wiley professional computing. J. Wiley. Chap. 10: Formative Evaluation, pp. 283–340 (cited on p. 40).

Hollnagel, Erik (1993a). "Human reliability analysis - Context and control". In: (cited on pp. 33, 102, 155).

— (July 1993b). "The phenotype of erroneous actions". In: *Int. J. Man-Mach. Stud.* 39.1, pp. 1–32. ISSN: 0020-7373. DOI: 10.1006/imms.1993.1051. URL: http://dx.doi.org/10.1006/imms.1993.1051 (cited on pp. 33, 102, 155).

Hornbæk, Kasper and Erik Frøkjaer (2008). "Comparison of techniques for matching of usability problem descriptions". In: *Interact. Comput.* 20.6, pp. 505–514. ISSN: 0953-5438. DOI: http://dx.doi.org/10.1016/j.intcom.2008.08.005 (cited on p. 59).

Hornbæk, Kasper and Erik Frøkjær (2008). "A Study of the Evaluator Effect in Usability Testing". In: *Human–Computer Interaction* 23.3, pp. 251–277. URL: http://www.informaworld.com/10.1080/07370020802278205 (cited on pp. 1, 16, 50, 55, 59, 67, 128, 224, 226).

Howarth, Jonathan Randall (Apr. 2007). "Supporting Novice Usability Practitioners with Usability Engineering Tools". Ph.D. thesis. Blacksburg, Virginia (cited on p. 37).

Hsieh, Hsiu-Fang and Sarah E. Shannon (2005). "Three Approaches to Qualitative Content Analysis". In: *Qualitative Health Research* 15.9, pp. 1277–1288. DOI: 10.1177/1049732305276687. eprint: http://qhr.sagepub.com/content/15/9/1277.full.pdf+html. URL: http://qhr.sagepub.com/content/15/9/1277.abstract (cited on p. 106).

Huizinga, Johan (1955). *Homo Ludens: A study of the play element in culture*. Beacon Press, Boston (cited on p. 4).

Hutchins, Edwin L., James D. Hollan, and Donald A. Norman (Dec. 1985). "Direct manipulation interfaces". In: *Hum.-Comput. Interact.* 1.4, pp. 311–338. ISSN: 0737-0024. DOI: 10.1207/s15327051hci0104_2. URL: http://dx.doi.org/10.1207/s15327051hci0104_2 (cited on p. 36).

Hvannberg, Ebba Thora, Effie Lai-Chong Law, and Marta Kristín Lárusdóttir (2007). "Heuristic evaluation: Comparing ways of finding and reporting usability problems". In: *Interact. Com-*

put. 19.2, pp. 225–240. ISSN: 0953-5438. DOI: http://dx.doi.org/10.1016/j.intcom.2006.10.001 (cited on p. 59).

Ijsselsteijn, W, Y de Kort, K Poels, and A Jurgelionis (Jan. 2007). "Characterising and Measuring User Experiences in Digital Games". In: *International Conference on Advances in Computer Entertainment*. URL: http://www.yvonnedekort.nl/pdfs/ACE%2525202007%252520workshop%252520submission%252520TUe%252520final.pdf (cited on pp. 4, 31, 38).

ISO (1998). *ISO/IEC 9241-11 Ergonomic Requirements for Office Work with Visual Display Terminals (VDTS). Part 11: Guidance on Usability*. Tech. rep. ISO/IEC 9241-11. Geneva: International Organization for Standardization (cited on pp. 13, 18–19).

— (June 2001). *ISO/IEC 9126-1:2001 Software engineering – Product quality – Part 1: Quality model*. Tech. rep. 9126-1:2001. Geneva: International Organization for Standardization (cited on p. 19).

— (July 2010). *ISO 9241-210 Ergonomics of human-system interaction. Part 210: Human-centred design for interactive systems*. Tech. rep. 9241-210. Geneva: International Organization for Standardization (cited on p. 228).

ISO/IEC (Aug. 2005). *ISO/IEC 25000:2005 Software Engineering – Software product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE*. Tech. rep. 25000:2005. Geneva: International Organization for Standardization / International Electro technical Commission (cited on p. 19).

Jacobsen, Niels Ebbe, Morten Hertzum, and Bonnie E. John (1998a). "The Evaluator Effect in Usability Studies: Problem Detection and Severity Judgments". In: *Human Factors and Ergonomics Society Annual Meeting Proceedings* 42.19, 1336–1340(5). URL: http://www.ingentaconnect.com/content/hfes/hfproc/1998/00000042/00000019/art00002 (cited on p. 1).

— (1998b). "The evaluator effect in usability tests". In: *CHI '98: CHI 98 conference summary on Human factors in computing systems*. Los Angeles, California, United States: ACM, pp. 255–256. ISBN: 1-58113-028-7. DOI: http://doi.acm.org/10.1145/286498.286737 (cited on pp. 1, 35, 52).

Järvinen, Aki (Feb. 2008). "Games without Frontiers Theories and Methods for Game Studies and Design". Ph.D. thesis. University of Tampere, Finland: Media Culture, pp. 1–416 (cited on pp. 23, 44).

Järvinen, Aki, Satu Heliö, and Frans Mäyrä (2002). *Communication and Community in Digital Entertainment Services - Prestudy Research Report*. Tech. rep. Finland: University of Tampere, Hypermedia Laboratory. URL: tampub.uta.fi/tup/951-44-5432-4.pdf (cited on pp. 4, 27, 224, 226).

Johnson, Daniel and John Gardner (2010). "Personality, Motivation and Video Games". In: *Proc. OzCHI* (cited on p. 23).

Jørgensen, A (Jan. 2004). "Marrying HCI/Usability and computer games: a preliminary look". In: *Proceedings of the third Nordic conference on Human-computer ...*. URL: http://portal.acm.org/citation.cfm?id=1028014.1028078 (cited on p. 22).

Juul, Jesper and Marleigh Norton (2009). "Easy to use and incredibly difficult: on the mythical border between interface and gameplay". In: *FDG '09: Proceedings of the 4th International Conference on Foundations of Digital Games*. Orlando, Florida: ACM, pp. 107–112. ISBN: 978-1-60558-437-9. DOI: http://doi.acm.org/10.1145/1536513.1536539 (cited on p. 22).

Kaiser, Henry F. (1960). "The Application of Electronic Computers to Factor Analysis". In: *Educational and Psychological Measurement* 20.1, pp. 141–151. DOI: 10.1177/001316446002000116. eprint: http://epm.sagepub.com/content/20/1/141.full.pdf+html. URL: http://epm.sagepub.com/content/20/1/141.short (cited on pp. 89–90).

Keenan, Susan L., H. Rex Hartson, Dennis G. Kafura, and Robert S. Schulman (1999). "The Usability Problem Taxonomy: A Framework for Classification and Analysis". In: *Empirical Softw.*

Engg. 4.1, pp. 71–104. ISSN: 1382-3256. DOI: http://dx.doi.org/10.1023/A:1009855231530 (cited on pp. 116, 228).

Kellar, Melanie, Carolyn Watters, and Jack Duffy (June 2005). "Motivational Factors in Game Play in Two User Groups". In: *Changing Views: Worlds in Play: Proceedings of the 2005 Digital Games Research Association Conference*. Ed. by de Castell Suzanne and Jenson Jennifer. Vancouver: University of Vancouver, p. 6. URL: http://www.digra.org/dl/display_html?chid=06278.15575.pdf (cited on p. 23).

Koivisto, Elina M.I. and Hannu Korhonen (2006). *Mobile Game Playability Heuristics*. Tech. rep. Nokia Research Center (cited on p. 10).

Komulainen, Jeppe, Jari Takatalo, Miikka Lehtonen, and Göte Nyman (2008). "Psychologically structured approach to user experience in games". In: *NordiCHI '08: Proceedings of the 5th Nordic conference on Human-computer interaction*. Lund, Sweden: ACM, pp. 487–490. ISBN: 978-1-59593-704-9. DOI: http://doi.acm.org/10.1145/1463160.1463226 (cited on p. 23).

Korhonen, Hannu (2010). "Comparison of playtesting and expert review methods in mobile game evaluation". In: *Proceedings of the 3rd International Conference on Fun and Games*. Fun and Games '10. Leuven, Belgium: ACM, pp. 18–27. ISBN: 978-1-60558-907-7. DOI: http://doi.acm.org/10.1145/1823818.1823820. URL: http://doi.acm.org/10.1145/1823818.1823820 (cited on pp. 68, 71).

Korhonen, Hannu, Janne Paavilainen, and Hannamari Saarenpää (2009). "Expert Review Method in Game Evaluations - Comparison of Two Playability Heuristic Sets". In: *Proc. Mindtrek* (cited on pp. 66, 68, 73, 85–86, 92, 103, 119).

Kotval, Xerxes P., Cheryl L. Coyle, Paulo A. Santos, Heather Vaughn, and Rebecca Iden (2007). "Heuristic evaluations at bell labs: analyses of evaluator overlap and group session". In: *CHI '07 extended abstracts on Human factors in computing systems*. CHI '07. San Jose, CA, USA: ACM, pp. 1729–1734. ISBN: 978-1-59593-642-4. DOI: http://doi.acm.org/10.1145/1240866.1240891. URL: http://doi.acm.org/10.1145/1240866.1240891 (cited on p. 67).

Krippendorff, Klaus. (2004). *Content analysis: an introduction to its methodology, 2nd ed.* Thousand Oaks, Calif: Sage. ISBN: 9780761915447; 9780761915454; 0761915443; 0761915451. URL: http://prism.talis.com/sussex-ac/items/851313 (cited on pp. 106, 225).

Laitinen, Sauli (Aug. 2008). "Usability and Playability Expert Evaluation". In: *Game Usability: Advancing the Player Experience*. Ed. by Katherine Isbister and Noah Schaffer. Morgan Kaufmann. Chap. 7, pp. 91–111 (cited on pp. 26, 28).

Landis, J R and G G Koch (Mar. 1977). "The measurement of observer agreement for categorical data". In: *Biometrics* 33.1, pp. 159–174. ISSN: 0006-341X (Print); 0006-341X (Linking) (cited on pp. 59, 223).

Lanzilotti, R., C. Ardito, M. F. Costabile, and A. De Angeli (Jan. 2011). "Do patterns help novice evaluators? A comparative study". In: *Int. J. Hum.-Comput. Stud.* 69 (1-2), pp. 52–69. ISSN: 1071-5819. DOI: http://dx.doi.org/10.1016/j.ijhcs.2010.07.005. URL: http://dx.doi.org/10.1016/j.ijhcs.2010.07.005 (cited on p. 59).

Lavery, Darryn, Gilbert Cockton, and Malcolm P. Atkinson (July 1997). "Comparison of evaluation methods using structured usability problem reports". In: *Behaviour and Information Technology* 16.21, pp. 246–266 (cited on pp. 31, 33–34, 97–98, 116, 223).

Law, Effie Lai-Chong and Ebba Thora Hvannberg (2004a). "Analysis of combinatorial user effect in international usability tests". In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. CHI '04. Vienna, Austria: ACM, pp. 9–16. ISBN: 1-58113-702-8. DOI: 10.1145/985692.985694. URL: http://doi.acm.org/10.1145/985692.985694 (cited on pp. 47, 59, 228).

— (2004b). "Analysis of strategies for improving and estimating the effectiveness of heuristic evaluation". In: *NordiCHI '04: Proceedings of the third Nordic conference on Human-computer interaction*. Tampere, Finland: ACM, pp. 241–250. ISBN: 1-58113-857-1. DOI: http://doi.acm.org/10.1145/1028014.1028051 (cited on pp. 55, 60, 67, 72, 144, 230).

Law, Effie Lai-Chong and Ebba Thora Hvannberg (2008). "Consolidating usability problems with novice evaluators". In: *Proceedings of the 5th Nordic conference on Human-computer interaction: building bridges*. NordiCHI '08. Lund, Sweden: ACM, pp. 495–498. ISBN: 978-1-59593-704-9. DOI: http://doi.acm.org/10.1145/1463160.1463228. URL: http://doi.acm.org/10.1145/1463160.1463228 (cited on pp. 16, 67).

Law, Effie Lai-chong and Ebba Thora Hvannberg (Sept. 2008). "Problems of Consolidating Usability Problems". In: *Proc. I-USED'*. http://www.scientificcommons.org/48840793. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.143.1167 (cited on p. 67).

Law, Effie Lai-Chong, Dominique Scapin, Dominique Scapin, Gilbert Cockton, Mark Springett, Christian Stary, and Marco Winckler (ed.s) (2009). "Maturation of Usability Evaluation Methods: Retrospect and Prospect (Final Reports of COST294-MAUSE Working Groups)". In: *COST294-MAUSE Closing Conference Proceedings*. COST. IRIT Press, Toulouse, France, p. 188 (cited on p. 41).

Law, Lai-Chong and Ebba Thora Hvannberg (2002). "Complementarity and convergence of heuristic evaluation and usability test: a case study of universal brokerage platform". In: *Proceedings of the second Nordic conference on Human-computer interaction*. NordiCHI '02. Aarhus, Denmark: ACM, pp. 71–80. ISBN: 1-58113-616-1. DOI: http://doi.acm.org/10.1145/572020.572030. URL: http://doi.acm.org/10.1145/572020.572030 (cited on pp. 42, 47).

Lazzaro, Nicole (Aug. 2008). "The Four Fun Keys". In: *Game Usability: Advancing the Player Experience*. Ed. by Katherine Isbister and Noah Schaffer. Morgan Kaufmann. Chap. 20, pp. 317–343 (cited on p. 29).

Lee, J. and C. Y. Im (2009). "A Study on User Centered Game Evaluation Guideline Based on the MIPA Framework". In: *Lecture Notes in Computer Science*. Ed. by M. Kurosu. Vol. 5619. Berlin: Springer-Verlag, pp. 84–93 (cited on p. 24).

Lewis, Clayton, Peter G. Polson, Cathleen Wharton, and John Rieman (1990). "Testing a walkthrough methodology for theory-based design of walk-up-and-use interfaces". In: *Proceedings of the SIGCHI conference on Human factors in computing systems: Empowering people*. CHI '90. Seattle, Washington, United States: ACM, pp. 235–242. ISBN: 0-201-50932-6. DOI: http://doi.acm.org/10.1145/97243.97279. URL: http://doi.acm.org/10.1145/97243.97279 (cited on p. 223).

Mahajan, Reenal (June 2003). "A Usability Problem Diagnosis Tool: Development and Formative Evaluation". MA thesis. Blacksburg, Virginia, USA: Virginia Polytechnic Institute and State University (cited on pp. 116, 228).

Malone, Thomas W. (1980). "What makes things fun to learn? heuristics for designing instructional computer games". In: *Proc. 3rd ACM SIGSMALL symposium and the first SIGPC symposium on Small systems*. Palo Alto, California, United States: ACM, pp. 162–169. ISBN: 0-89791-024-9. DOI: http://doi.acm.org/10.1145/800088.802839 (cited on pp. 23–24, 85).

— (1982). "Heuristics for designing enjoyable user interfaces: Lessons from computer games". In: *Proc. Conference on Human factors in computing systems*. Gaithersburg, Maryland, United States: ACM, pp. 63–68. DOI: http://doi.acm.org/10.1145/800049.801756 (cited on pp. 85, 114).

Malone, Thomas W. and M.R. Lepper (1987). "Making learning fun: a taxonomy of intrinsic motivation for learning". In: *Aptitude, learning and interaction III. Conative and affective process analysis*. Ed. by R.E. Snow and M.J. Farr. Aptitude, learning, and instruction. Hillsdale, NJ, USA: L. Erlbaum. Chap. 10, pp. 223–253 (cited on p. 23).

Matera, Maristella (1999). "SUE: A Systematic Methodology for Evaluating Hypermedia Usability". Ph.D. thesis. Dipartimento di Elettronica e Informazione, Politecnico Di Milano (cited on p. 124).

Matera, M., M.F. Costabile, F. Garzotto, and P. Paolini (Jan. 2002). "SUE inspection: an effective method for systematic usability evaluation of hypermedia". In: *Systems, Man and Cybernetics,*

Part A: Systems and Humans, IEEE Transactions on 32.1, pp. 93 –103. ISSN: 1083-4427. DOI: 10.1109/3468.995532 (cited on pp. 42, 54, 73, 105).

McAllister, Graham and Gareth R. White (2010). "Video Game Development and User Experience". In: *Evaluating User Experience in Games - Concepts and Methods*. Ed. by Regina Bernhaupt. Springer-Verlag. Chap. 7 (cited on pp. 45, 79).

Mentis, Helena and Geri Gay (2003). "User recalled occurrences of usability errors: implications on the user experience". In: *CHI '03: CHI '03 extended abstracts on Human factors in computing systems*. Ft. Lauderdale, Florida, USA: ACM, pp. 736–737. ISBN: 1-58113-637-4. DOI: http://doi.acm.org/10.1145/765891.765959 (cited on pp. 116, 228).

Meurs, Richard van (Aug. 2007). "How to Play the Game: a study on MUD player types and their real life personality traits". MA thesis. Tilburg University (cited on p. 23).

Molich, Rolf, Meghan R. Ede, Klaus Kaasgaard, and Barbara Karyukin (2004). "Comparative usability evaluation". In: *Behav. Inf. Technol.* 23.1, pp. 65–74. ISSN: 0144-929X. DOI: http://dx.doi.org/10.1080/01449290320001739 51 (cited on p. 1).

Monk, Andrew, Marc Hassenzahl, Mark Blythe, and Darren Reed (2002). "Funology: designing enjoyment". In: *CHI '02: CHI '02 extended abstracts on Human factors in computing systems*. Minneapolis, Minnesota, USA: ACM, pp. 924–925. ISBN: 1-58113-454-1. DOI: http://doi.acm.org/10.1145/506443.506661 (cited on p. 23).

Nacke, Lennart (2009). "From playability to a hierarchical game usability model". In: *FuturePlay '09: Proceedings of the 2009 Conference on Future Play on @ GDC Canada*. Vancouver, British Columbia, Canada: ACM, pp. 11–12. ISBN: 978-1-60558-685-4. DOI: http://doi.acm.org/10.1145/1639601.1639609 (cited on p. 27).

Nielsen, Jakob (1992). "Finding usability problems through heuristic evaluation". In: *CHI '92: Proceedings of the SIGCHI conference on Human factors in computing systems*. Monterey, California, United States: ACM, pp. 373–380. ISBN: 0-89791-513-5. DOI: http://doi.acm.org/10.1145/142750.142834 (cited on p. 87).

— (1994a). "Enhancing the explanatory power of usability heuristics". In: *CHI '94: Proceedings of the SIGCHI conference on Human factors in computing systems*. Boston, Massachusetts, United States: ACM, pp. 152–158. ISBN: 0-89791-650-6. DOI: http://doi.acm.org/10.1145/191666.191729 (cited on pp. 10, 64, 84–86, 88, 93, 114, 148).

— (1994b). "Estimating the number of subjects needed for a thinking aloud test". In: *International Journal of Human-Computer Studies* 41.3, pp. 385 –397. ISSN: 1071-5819. DOI: 10.1006/ijhc.1994.1065. URL: http://www.sciencedirect.com/science/article/pii/S1071581984710652 (cited on pp. 32, 62, 129).

— (1994c). "Usability inspection methods". In: *Conference companion on Human factors in computing systems*. CHI '94. Boston, Massachusetts, United States: ACM, pp. 413–414. ISBN: 0-89791-651-4. DOI: http://doi.acm.org/10.1145/259963.260531. URL: http://doi.acm.org/10.1145/259963.260531 (cited on pp. 41, 187).

Nielsen, Jakob and Rolf Molich (1990). "Heuristic evaluation of user interfaces". In: *Proc. SIGCHI*. Seattle, Washington, United States: ACM, pp. 249–256. ISBN: 0-201-50932-6. DOI: http://doi.acm.org/10.1145/97243.97281 (cited on pp. 62, 150).

Nielsen Norman Group (2011). *User Experience - Our Definition*. URL: http://www.nngroup.com/about/userexperience.html (cited on p. 228).

Norman, Donald A. (1986). "Cognitive Engineering". In: *User centered system design;new perspectives on human-computer interaction*. Ed. by Donald A. Norman and Stephen W. Draper. Hillsdale, NJ, USA: Lawrence Erlbaum Associates Inc. Chap. 3, pp. 31–61 (cited on pp. 36–37, 45, 116, 155, 228).

Omar, Hasiah Mohamed and Azizah Jaafar (2009). "Conceptual Framework for a Heuristics Based Methodology for Interface Evaluation of Educational Games". In: *Computer Technology and Development, International Conference on* 1, pp. 594–598. DOI: http://doi.ieeecomputersociety.org/10.1109/ICCTD.2009.249 (cited on p. 25).

Pagulayan, Randy J., Kevin Keeker, Dennis Wixon, Ramon L. Romero, and Thomas Fuller (2003). "User-centered design in games". In: *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications*. Ed. by Julie A. Jacko and Andrew Sears. Hillsdale, NJ, USA: L. Erlbaum Associates Inc. Chap. User-centered design in games, pp. 883–906. ISBN: 0-8058-3838-4. URL: http://portal.acm.org/citation.cfm?id=772072.772128 (cited on p. 36).

Pausch, R, R Gold, T Skelly, and D Thiel (Jan. 1994). "What HCI designers can learn from video game designers". In: *Conference on Human Factors in Computing Systems*. URL: http://portal.acm.org/citation.cfm?id=260220 (cited on p. 23).

Pinelle, David, Nelson Wong, and Tadeusz Stach (2008a). "Heuristic evaluation for games: usability principles for video game design". In: *Proc. SIGCHI*. Florence, Italy: ACM, pp. 1453–1462. ISBN: 978-1-60558-011-1. DOI: http://doi.acm.org/10.1145/1357054.1357282 (cited on pp. 66, 68, 71, 85–86, 103, 119).

— (2008b). "Using genres to customize usability evaluations of video games". In: *Future Play '08: Proceedings of the 2008 Conference on Future Play*. Toronto, Ontario, Canada: ACM, pp. 129–136. ISBN: 978-1-60558-218-4. DOI: http://doi.acm.org/10.1145/1496984.1497006 (cited on p. 92).

Polson, Peter G. and Clayton H. Lewis (June 1990). "Theory-based design for easily learned interfaces". In: *Hum.-Comput. Interact.* 5 (2), pp. 191–220. ISSN: 0737-0024. DOI: http://dx.doi.org/10.1207/s15327051hci0502\&3_3. URL: http://dx.doi.org/10.1207/s15327051hci0502\&3_3 (cited on p. 116).

Polson, Peter G., Clayton Lewis, John Rieman, and Cathleen Wharton (May 1992). "Cognitive walkthroughs: a method for theory-based evaluation of user interfaces". In: *Int. J. Man-Mach. Stud.* 36 (5), pp. 741–773. ISSN: 0020-7373. DOI: 10.1016/0020-7373(92)90039-N. URL: http://dl.acm.org/citation.cfm?id=141744.141755 (cited on pp. 103, 116).

Rasmussen, Jens (1982). "Human errors. A taxonomy for describing human malfunction in industrial installations". In: *Journal of Occupational Accidents* 4.2–4, pp. 311 –333. ISSN: 0376-6349. DOI: 10.1016/0376-6349(82)90041-4. URL: http://www.sciencedirect.com/science/article/pii/0376634982900414 (cited on p. 36).

Rebellion Developments (Feb. 2010). *Aliens vs. Predator* (cited on p. 84).

Rigby, Scott and Richard Ryan (Jan. 2007a). "Rethinking Carrots: A New Method For Measuring What Players Find Most Rewarding and Motivating About Your Game". In: *Gamasutra* (cited on p. 23).

— (Sept. 2007b). "The Player Experience of Need Satisfaction (PENS) An applied model and methodology for understanding key components of the player experience". In: (cited on p. 23).

Robin, Hunicke, Leblanc Marc, and Zubek Robert (2004). "MDA: A Formal Approach to Game Design and Game Research". In: *Proc. Challenges in Game AI Workshop, Nineteenth National Conference on Artificial Intelligence (AAAI '04)*. San Jose, California: AAAI Press. URL: http://cs.northwestern.edu/~hunicke/pubs/MDA.pdf (cited on p. 24).

Rohn, Janice A., Jared Spool, Mayuresh Ektare, Sanjay Koyani, Michael Muller, and Janice (Ginny) Redish (2002). "Usability in practice: alternatives to formative evaluations-evolution and revolution". In: *CHI '02 extended abstracts on Human factors in computing systems*. CHI EA '02. Minneapolis, Minnesota, USA: ACM, pp. 891–897. ISBN: 1-58113-454-1. DOI: http://doi.acm.org/10.1145/506443.506648. URL: http://doi.acm.org/10.1145/506443.506648 (cited on p. 72).

Rosson, Mary Beth and John M. Carroll (2003). "The human-computer interaction handbook". In: ed. by Julie A. Jacko and Andrew Sears. Hillsdale, NJ, USA: L. Erlbaum Associates Inc. Chap. Scenario-based design, pp. 1032–1050. ISBN: 0-8058-3838-4. URL: http://dl.acm.org/citation.cfm?id=772072.772137 (cited on pp. 118, 227).

Rosson, M.B. and J.M. Carroll (2002). *Usability Engineering: Scenario-Based Development of Human-Computer Interaction*. Morgan Kaufmann Series in Interactive Technologies. Academic Press. ISBN: 9781558607125. URL: http://books.google.co.uk/books?id=sRPg0IYhYFYC (cited on p. 40).

Ryan, Richard, C. Scott Rigby, and Andrew Przybylski (Jan. 2006). "The motivational pull of video games: A self-determination theory approach". In: *Motivation and Emotion* (Volume 30, Number 4 / December, 2006), pp. 344–360. ISSN: 0146-7239 (Print) 1573-6644 (Online). DOI: 10.1007/s11031-006-9051-8. URL: http://www.springerlink.com/index/H8U63440VL4Q6534.pdf (cited on p. 23).

Sauro, Jeff and Erika Kindlund (2005). "A method to standardize usability metrics into a single score". In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. CHI '05. Portland, Oregon, USA: ACM, pp. 401–409. ISBN: 1-58113-998-5. DOI: http://doi.acm.org/10.1145/1054972.1055028. URL: http://doi.acm.org/10.1145/1054972.1055028 (cited on p. 57).

Schaffer, Noah (Apr. 2007). *Heuristics for Usability in Games*. Tech. rep. PlayerFriendly.com (cited on p. 85).

— (Apr. 2009). "Verifying An Integrated Model Of Usability In Games". Ph.D. thesis. Rensselaer Polytechnic Institute, Troy, New York: Dept. of Language, Literature, et al. (cited on pp. 23, 29).

Schmettow, Martin and Sabine Niebuhr (2007). "A pattern-based usability inspection method: first empirical performance measures and future issues". In: *BCS-HCI '07: Proceedings of the 21st British HCI Group Annual Conference on People and Computers*. University of Lancaster, United Kingdom: British Computer Society, pp. 99–102. ISBN: 978-1-902505-95-4 (cited on p. 96).

Schuurman, Dimitri, Katrien De Moor, Lieven De Marez, and Jan Van Looy (2008). "Fanboys, competers, escapists and time-killers: a typology based on gamers' motivations for playing video games". In: *DIMEA '08: Proceedings of the 3rd international conference on Digital Interactive Media in Entertainment and Arts*. Athens, Greece: ACM, pp. 46–50. ISBN: 978-1-60558-248-1. DOI: http://doi.acm.org/10.1145/1413634.1413647 (cited on p. 23).

Scriven, Michael (1967). "The Methodology of Evaluation". In: *Perspectives of curriculum evaluation*. Ed. by Ralph W Tyler, Robert M Gagné, and Michael Scriven. Chicago: Rand McNally. Chap. The Methodology of Evaluation (cited on pp. 39–40, 45–46, 53, 61–62, 67, 74, 81).

Sears, Andrew (1997). "Heuristic Walkthroughs: Finding the Problems Without the Noise". In: *International Journal of Human-Computer Interaction* 9.3, pp. 213–234. URL: http://www.informaworld.com/10.1207/s15327590ijhc0903_2 (cited on pp. 57, 60, 230).

Shackel, Brian (2009). "Usability - Context, framework, definition, design and evaluation". In: *Interacting with Computers* 21.5-6, pp. 339 –346. ISSN: 0953-5438. DOI: DOI:10.1016/j.intcom.2009.04.007. URL: http://www.sciencedirect.com/science/article/B6V0D-4W99VWW-1/2/f8f10dd5b43df2bd66d26adfed437440 (cited on p. 20).

Sherry, John and Kristen Lucas (May 2003). "Video Game Uses and Gratifications as Predictors of Use and Game Preference". In: *Proceedings of International Communication Association 2003*. Marriott Hotel, San Diego, CA (cited on p. 23).

Sim, Gavin R. (2009). "Evidence Based Design of Heuristics: Usability and Computer Assisted Assessment". Ph.D. thesis. Preston: School of Computing, Engineering and Physical Sciences, University of Central Lancashire (cited on pp. 51, 67, 70, 72).

Sim, Gavin and Janet C. Read (2010). "The Damage Index: an aggregation tool for usability problem prioritisation". In: *Proceedings of the 24th BCS Interaction Specialist Group Conference*. BCS '10. Dundee, United Kingdom: British Computer Society, pp. 54–61. ISBN: 978-1-78017-130-2. URL: http://dl.acm.org/citation.cfm?id=2146303.2146311 (cited on p. 67).

Somervell, Jacob (June 2004). "Developing heuristic evaluation methods for large screen information exhibits based on critical parameters". Ph.D. thesis. Virginia Polytechnic Institute and State University (cited on p. 118).

Springett, Mark (1998). "Linking surface error characteristics to root problems in user-based evaluation studies". In: *Proceedings of the working conference on Advanced visual interfaces*. AVI '98. L'Aquila, Italy: ACM, pp. 102–113. DOI: 10.1145/948496.948512. URL: http://doi.acm.org/10.1145/948496.948512 (cited on pp. 29, 116).

Sridharan, Sriram (Dec. 2001). "Usability and Reliability of the User Action Framework: A Theoretical Foundation for Usability Engineering Activities". MA thesis. Department of Industrial and Systems Engineering (cited on pp. 116, 228).

Stevens, J.P. (2002). *Applied Multivariate Statistics for the Social Sciences, Fifth Edition*. Applied Multivariate STATS. Taylor & Francis. ISBN: 9780805837766. URL: http://books.google.co.uk/books?id=mK0MtyWa7-QC (cited on p. 90).

Tech, HCI Virginia (Mar. 2012). *The Concept of Affordance in Usability*. URL: http://research.cs.vt.edu/usability/projects/uaf%20and%20tools/affordance.htm (cited on p. 223).

Theofanos, Mary and Whitney Quesenbery (Aug. 2005). "Reporting on Formative Testing: A UPA 2005 Workshop Report". In: *Journal of Usability Studies* 1, pp. 27–45 (cited on p. 40).

Thompson, Jennifer A. (Dec. 1999). "Investigating the Effectiveness of Applying the Critical Incident Technique to Remote Usability Evaluation". MA thesis. Virginia Polytechnic Institute and State University: Industrial and Systems Engineering (cited on p. 37).

Tychsen, Anders, Michael Hitchens, and Thea Brolund (2008). "Motivations for play in computer role-playing games". In: *Future Play '08: Proceedings of the 2008 Conference on Future Play*. Toronto, Ontario, Canada: ACM, pp. 57–64. ISBN: 978-1-60558-218-4. DOI: http://doi.acm.org/10.1145/1496984.1496995 (cited on p. 23).

Vermeeren, Arnold Petrus Otto Sabina, Ilse E.H. van Kesteren, and Mathilde M. Bekker (2003). "Managing the Evaluator Effect in User Testing". In: *Proc. HCI Interact*. Ed. by M. Rauterberg et al. IOS Press, pp. 674–654 (cited on pp. 3, 54, 59, 224).

Vermeeren, Arnold P.O.S., Jelle Attema, Evren Akar, Huib de Ridder, Andrea J. von Doorn, Cigdem Erbug, Ali E. Berkman, and Martin C. Maguire (2008). "Usability Problem Reports for Comparative Studies: Consistency and Inspectability". In: *Human–Computer Interaction* 23.4, pp. 329–380. DOI: 10.1080/07370020802536396. eprint: http://www.tandfonline.com/doi/pdf/10.1080/07370020802536396. URL: http://www.tandfonline.com/doi/abs/10.1080/07370020802536396 (cited on pp. 55, 59).

Vermeeren, Arnold P. O. S., Karin den Bouwmeester, Jans Aasman, and Huib de Ridder (2002). "DEVAN: a tool for detailed video analysis of user test data". In: *Behaviour & Information Technology* 21.6, pp. 403–423. URL: http://www.informaworld.com/10.1080/0144929021000051714 (cited on pp. 33, 54, 224).

Vorderer, Peter, Tilo Hartmann, and Christoph Klimmt (2003). "Explaining the enjoyment of playing video games: the role of competition". In: *ICEC '03: Proceedings of the second international conference on Entertainment computing*. Pittsburgh, Pennsylvania: Carnegie Mellon University, pp. 1–9 (cited on p. 23).

Ward, Liam (Aug. 2010). *A Phenomenology Investigation Into The Motivations of Video Game Use*. URL: http://socyberty.com/philosophy/a-phenomenology-investigation-into-the-motivations-of-video-game-use/ (cited on p. 23).

Welie, Martijn Van, Gerrit C. Van Der Veer, and Anton Eliëns (1999). "Breaking down Usability". In: *Proceedings of Interact '99*. Press, pp. 613–620 (cited on p. 32).

Whitton, Nicola (2007). "Motivation and computer game based learning". In: *Proceedings of ASCILITE 2007* (cited on p. 23).

Winter, Sebastian, Stefan Wagner, and Florian Deissenboeck (2008). "A Comprehensive Model of Usability". In: *Engineering Interactive Systems*. Ed. by Jan Gulliksen, Morton Harning, Philippe

Palanque, Gerrit van der Veer, and Janet Wesson. Vol. 4940. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 106–122 (cited on p. 116).

Wixon, Dennis (July 2003). "Evaluating usability methods: why the current literature fails the practitioner". In: *interactions* 10 (4), pp. 28–34. ISSN: 1072-5520. DOI: http://doi.acm.org/10.1145/838830.838870. URL: http://doi.acm.org/10.1145/838830.838870 (cited on p. 41).

Woolrych, Alan, Gilbert Cockton, and Mark Hindmarch (2004). "Falsification Testing For Usability Inspection Method Assessment". In: *Proceedings of HCI 2004 Conference on People and Computers XVIII*. Ed. by S. Fincher, P. Markopoulos, D.§ Moore, and R. Ruddle. BCS. Bath (cited on p. 49).

Woolrych, Alan, Kasper Hornbæk, Erik Frøkjær, and Gilbert Cockton (2011). "Ingredients and Meals Rather Than Recipes: A Proposal for Research That Does Not Treat Usability Evaluation Methods as Indivisible Wholes". In: *International Journal of Human-Computer Interaction* 27.10, pp. 940–970. DOI: 10.1080/10447318.2011.555314. eprint: http://dx.doi.org/10.1080/10447318.2011.555314. URL: http://dx.doi.org/10.1080/10447318.2011.555314 (cited on p. 152).

Yee, Nick (2006a). "Motivations for play in online games". In: *Cyberpsychology & behavior* 9.6, pp. 772–775 (cited on p. 23).

— (2006b). "The demographics, motivations, and derived experiences of users of massively multi-user online graphical environments". In: *Presence: Teleoper. Virtual Environ.* 15.3, pp. 309–329. ISSN: 1054-7460. DOI: http://dx.doi.org/10.1162/pres.15.3.309 (cited on p. 23).

Yue, Wong Seng and Nor Azan Mat Zin (2009). "Usability evaluation for history educational games". In: *ICIS '09: Proceedings of the 2nd International Conference on Interaction Sciences*. Seoul, Korea: ACM, pp. 1019–1025. ISBN: 978-1-60558-710-3. DOI: http://doi.acm.org/10.1145/1655925.1656110 (cited on p. 27).

Zammitto, Veronica Lorena (2010). "Gamers' Personality And Their Gaming Preferences". MA thesis. School of Interactive Arts and Technology, Simon Fraser University (cited on p. 23).

Zapf, Dieter, Felix C. Brodbeck, Michael Frese, Helmut Peters, and Jochen Prümper (1992). "Errors in working with office computers: A first validation of a taxonomy for observed errors in a field setting". In: *International Journal of Human-Computer Interaction* 4.4, pp. 311–339. URL: http://www.informaworld.com/10.1080/10447319209526046 (cited on p. 36).

Glossary

Any-Two a measure of agreement between two or more raters of usability data, expressed as a percentage from 0% for absolutely no agreement to 100% for complete agreement between all evaluators. Defined as, "...the number of problems two evaluators have in common divided by the number of problems they collectively detect, averaged over all possible pairs of two evaluators." (Hertzum and Jacobsen, 2001) Defined in Equation (eq. 3). 50, 54, 58, 59, 61, 128, 131, 133, 135, 136, 140, 142, 149, 150, 225, 230, *c.f.* Cohen's Kappa & Krippendorff's Alpha

breakdown (interaction breakdown) a behavioural or cognitive action that is incorrect or undesirable according to the designer or evaluator's expectation of the correct or desirable sequence of actions. For example, the user presses the wrong button, does not notice a status indicator, or misunderstands the intention of a design feature Lavery et al. (1997). 3, 10, 11, 31–34, 48, 54, 81, 98, 102, 104–112, 115, 116, 118–124, 126, 127, 135, 140, 172–176, 178–184, 226, 227, *see* outcome

cognitive affordance "a design feature that helps, aids, supports, facilitates, or enables thinking and/or knowing about something." (Hartson, 2003; Tech, 2012)
In the context of a first-person shooter game, cognitive affordances are diverse and might include indicators to show the player character's current health, the head-up display, instructions for how to use a skill or feature, etc. 127

cognitive playthrough a structured evaluation methodology, based on cognitive walkthrough, but for use with video game evaluation (Barr, 2010a,b). *see* playthrough evaluation & cognitive walkthrough

cognitive task analysis an analytical approach to understanding and describing the cognitive processes involved in tasks performance. *c.f.* hierarchical task analysis

cognitive walkthrough a structured evaluation methodology based on cognitive theories of learning. Characterised by very precise interaction analysis, often described at the level of individual key strokes. A "correct" or optimal interaction sequence is defined in advance of user testing, and deviations in the users' interaction are noted as problems. Lewis et al. (1990). 1, 6, 7, 14, 20, 33, 40, 41, 46–48, 50, 51, 103, 117, 150, 223, 226, 227, *see* usability evaluation method, usability inspection method, cognitive playthrough & playthrough evaluation

Cohen's Kappa a statistical measure of inter-rater agreement, often used for inter-evaluator reliability with a known, fixed set of data points (Cohen, 1960). Typical interpretations (Landis and Koch, 1977) for values are:
<0.00 Poor
0.00-0.20 Slight
0.21-0.40 Fair

0.41-0.60 Moderate
0.61-0.80 Substantial
0.81-1.00 Almost Perfect
Greve and Wentura suggest interpreting scores $\kappa < .4$ as "not be taken too seriously" and values of $0.4 \leq \kappa < 0.6$ as acceptable. $0.75 \leq \kappa$ seems good up to excellent. 52, 58–60, 117, 128, 225, *c.f.* Any-Two & Krippendorff's Alpha

DEtailed Video ANalysis a methodological tool for the transcription and evaluation of user test video footage A. P. O. S. Vermeeren et al. (2002). 54, 229

Discovery and Analysis REsources a model describing the properties of a usability evaluation method to discovery and analyse usability issues Cockton, Woolrych, Hall, et al. (2003). 110, 229

evaluator effect an effect seen in usability evaluations where the individual evaluator plays a significant role in the evaluation, and where different evaluators produce different results when evaluating the same system. i.e., the methodology is prone to issues of evaluator experience, subjectivity, and even cultural background. Defined by A. P. O. S. Vermeeren et al. (2003) as limited agreement in identified problems between multiple evaluators' analysis of the outcomes of a single user test. Defined by Hertzum and Jacobsen (2001) as "differences in evaluators' problem detection and severity ratings". Hornbæk and Frøkjær (2008) describe it as when "usability evaluators report substantially different sets of usability problems". vi, 1–3, 12–16, 29, 47, 48, 50, 52, 54–56, 60, 76–78, 96, 102, 109, 110, 117, 128, 136, 137, 140, 141, 143, 145, 147, 148, 150, 155, 226, *c.f.* user effect

first-person shooter an action video game genre, characterised by fast paced combat, seen from the perspective of the player character. e.g., *Half Life*, *Quake*, *Call of Duty*. v, vi, 3–7, 9, 13–15, 20–22, 31, 35, 37, 38, 46, 47, 75, 77–79, 81–83, 85, 87, 92–94, 96, 101, 104, 109, 117, 131, 137, 138, 145, 147–149, 151, 153, 223, 226–228

formative a formative evaluation method is applied during the formation of a product, especially early in the development lifecycle, and are intended to inform the ongoing design process. They do not necessarily require a running system, and could for example use paper prototypes or formal analysis of an early design specification instead. 2, 3, 11, 30, 39–45, 60, 62, 67, 68, 74, 77, 79, 80, 103, 115, 118, 144, 145, 148, 153, *c.f.* summative

gameplay noun, referring to the mechanics and dynamics of the game. In contrast to the verb "game play", which refers to playing games without the specific definitions of gameplay. Whereas playability refers to the potential for play, gameplay refers to the actual experience in action, though it is still generally used in a system-centric, essentialist way (Cockton, 2012). Järvinen et al. (2002) use gameplay to refer to "the time period during which a game imposes its rules and its environment on the player". 2, 21–25, 27, 28, 30, 31, 35, 38, 44, 71, 79, 80, 85, 155, 183, *c.f.* playability & player experience

Goals, Operators, Methods, Selection rules a formal model to analyse expert behaviour (Card et al., 1983). 227, 229

head-up display also known as Heads-Up Display. Visualisation of status information on the main display screen. For example, items such as a small map, ammunition count, and character health status are often displayed in one corner of the screen to provide the player with a quick and easy overview of their location, character equipment, and health status. 37, 115, 118, 121, 122, 223, 228

heuristic evaluation a usability inspection method. Typically lists of 10 items are used by an expert evaluator to help guide the evaluation. vi, 1, 2, 5–7, 9–12, 14–16, 21, 33, 39–45, 47, 49–51, 58, 62–65, 67–75, 77, 78, 81, 83–85, 87, 89, 91–98, 101, 102, 104, 110–112, 115, 117, 118, 123, 125, 128–131, 135, 137–140, 142–144, 147–151, 153, 186, 187, 190, 226, 227, 232

hierarchical task analysis an analytical method to understand and describe task performance in a hierarchical structure. Particularly emphasising goal-oriented tasks, the actions necessary to achieve them, and the conditions for their completion. *c.f.* cognitive task analysis

human-computer interaction a discipline of study, interested in the interaction between humans and computer systems, particularly with respect to design and evaluation. iv, 1, 4, 7, 19, 22, 23, 43–45, 76, 77, 86, 225, 226

inter-evaluator reliability a measure of agreement between two or more evaluators of usability data. In the case of comparisons between different usability evaluation methods, simple inter-rater reliability is usually not possible as rating scales will not be the same across each method. Instead subjective judgements must be made as to whether issues identified by the different methods represent the same underlying problem or not. 9, 14, 46, 50, 54, 62, 63, 66–68, 70, 84, 87, 94, 97, 98, 102, 109, 111, 112, 129, 131, 135, 136, 138, 140, 142, 148, 149, 223, 225, *see* inter-rater reliability

inter-rater reliability a measure of inter-evaluator reliability between two or more raters of usability data. Data usually consists of video footage of user test sessions, or expert analysis. Evaluators' ratings may be to assign a severity level to describe the magnitude of a usability problem, or to rate how well a heuristic explains an issue, etc. Ratings are usually made using ordinal scales, with perhaps five points. Typical algorithms for computing reliability include Any-Two agreement, Krippendorff's Alpha and Cohen's Kappa. Intra-rater reliability is measured the same way, but with a single evaluator rating the same data on two or more separate occasions. 9, 52, 70, 72, 81, 83, 84, 86, 88, 91, 93, 95, 97, 98, 103, 105, 108, 110, 112, 117, 128, 142, 145, 148, 225, *see* inter-evaluator reliability

keystroke-level model a formal model to measure and predict the time taken for an expert to perform atomic actions while using a computer system (Card et al., 1980). 46, *see* Goals, Operators, Methods, Selection rules

Krippendorff's Alpha a statistical measure of inter-rater agreement, often used for inter-evaluator reliability with multiple evaluators (Hayes and Krippendorff, 2007). Krippendorff (2004) cautiously suggests interpretations for values:
"When agreement is observed to be perfect ... $\alpha = 1$, indicating perfect reliability. When agreement and disagreement are matters of chance ... $\alpha = 0$, indicating the absence of reliability."
"Rely only on variables with reliabilities above $\alpha = .800$. Consider variables with reliabilities between $\alpha = .667$ and $\alpha = .800$ only for drawing tentative conclusions" . 83, 86, 93, 97, 148, 225, *c.f.* Any-Two & Cohen's Kappa

ludology a relatively modern academic discipline, emerging during the late 1990s. Ludology is the study of (video) games, and argues for the development of novel techniques and approaches specific to the medium instead of naïvely reusing existing methods from non-specialised disciplines (although they may have a role to play). For example, narratology is well positioned to address games through a perspective of narrative, and human-computer interaction for the challenges of use and interaction between two such

disparate entities of human and computer. Neither have the specific resources to address games *as games* however. Frasca (2001) provides the definition, "Ludology is the discipline that studies games... ludology studies games and playing in general, leaving videogames a just a particular branch of study". 44

matcher effect a subset of the evaluator effect, defined by Hornbæk and Frøkjær (2008) as "the difference between persons' matchings", and where the term "matching" is, "the procedure used for comparing usability problems found by different evaluators to assess whether they concern the same or different problems". This effect is largely overlooked in the literature as most studies do not provide procedures for this stage of evaluation, relying instead on purely subjective interpretation. 16, 50, 128, *c.f.* evaluator effect, user effect & wildcard effect

non-player character a term that refers to a game character that is controlled by the computer rather than by a human player. 23, 35, 37, *c.f.* player character

outcome (interaction outcome) a usability effect caused by an interaction breakdown. Defined in terms of the three principal usability aspects, Effectiveness, Efficiency, and Satisfaction. For example, if the user misunderstood a button in the interface (a breakdown), and wasted time by clicking it, Efficiency may be impacted (the outcome). 14, 15, 32, 34, 42, 73, 98, 102–106, 109–112, 115, 116, 118–122, 124, 127, 135, 172–174, 180–182, *see* breakdown

playability how playable a game is, comparable to how "usable" a game is, but additionally including aesthetics of play. Implies a system-centric, essentialist perspective, as if this were a property of the system itself (Cockton, 2012). Generally concerned with mechanics, dynamics, learning curve, replayability, aesthetics etc. Järvinen et al. (2002) explicitly distinguishes between four types of playability: Social; Functional; Structural; Audiovisual. Further detailed in Chapter 2 (Literature Review). 11, 12, 21, 22, 25, 27–31, 44, 68, 71, 80, 85, 140, 154, 155, 224, *see* usability, user experience & player experience

player action framework based on the user action framework, this is a tree hierarchy for categorising and analysing problem breakdowns during usability evaluation of video games. 10, 11, 61, 99, 109, 126, 127, 135, 136, 139, 144, 152, 154–156, 172, 176, 182, 186, 188, 190

player character a term that refers to the combination of human player and their virtual character in the game. A character that is controlled by a player cannot meaningfully be said to "be" without the player's control. When referring to the human only (for example when describing physically pressing buttons) the appropriate term is player. In contrast it would be incorrect to say that the "player" killed the enemy, as the player only used the physical controls that made the player character kill the enemy. When referring only to the visual representation of the character the term is avatar. The term character itself refers to not only the avatar, but also the game state, and any related narrative backstory, for example. 37, 121, 122, 223, 226, *c.f.* non-player character

player experience while user experience deals with the *use* of interactive systems, framed within an ecology of computer technology (i.e., human-computer interaction), player experience addresses the qualities of play, framed within an ecology of games (i.e., Game-Player Interaction; ludology). For example, in a traditional productivity application, usability issues are entirely discouraged, though they may add something important to the exploration involved in a video game as a certain amount of struggle can actually improve the player experience. This thesis limits itself to the detection of usability issues, and does not consider when usability issues are acceptable and when they are not. 12, 13, 20, 22, 26, 27, 57, 62, 74, 77, 79–81, 92, 109, 115, 136, 140, 141, 154, 155, 184, 226

playthrough evaluation the novel usability evaluation method developed in this thesis. Inspired by cognitive walkthrough and heuristic evaluation, but adapted for first-person shooter games. vi, 1, 5–7, 10, 11, 32, 42, 46, 48, 49, 51–54, 56, 66, 70, 78, 81, 82, 94, 97, 98, 109–113, 115, 118, 124, 125, 128–133, 135–146, 149–153, 183, 184, 227, 232

playthrough evaluation framework defines a framework in which playthrough evaluation can be used as a usability evaluation method, or as a means to evaluate and train evaluators. vi, 11, 109, 110, 112, 113, 115, 117, 118, 121, 122, 129–132, 137, 140, 144, 145, 150–155, 232

principal components analysis is a statistical method for revealing underlying or latent similarities between variables. 9, 11, 81, 84, 87–93, 103, 110, 112, 114, 115, 119, 148, 158

real-time strategy is a video game genre, typically simulating military battles. e.g., *Starcraft*. 229

role-playing game is a video game genre, often set in fantasy worlds and featuring an emphasis on character and narrative. e.g., *Baldur's Gate*. 229

scenario-based design a user-centered approach to design and evaluation. A scenario is a textual description of a user interaction. Scenarios are a narrative description of a hypothetical user in a specific context, using a concrete design for a defined reason. The design elements mentioned in the description are analysed by the "claims" that they make for the user experience. These are typically one sentence statements describing either a positive or negative potential outcome for the user (Carroll, 1995; M. B. Rosson and J. M. Carroll, 2003). 10, 118

summative evaluation that is intended to produce a summary assessment of a product's usability. Requires a running system, usually a finished product. 2, 3, 5, 30, 39, 40, 42–46, 58, 60, 62, 67, 68, 74, 76–79, 81, 83, 84, 101, 103, 115, 118, 145, 148, 153, *c.f.* formative

think aloud a technique employed in user test, where participants are asked to verbalise their thought process while interacting with the system. This can prove difficult in situations of high cognitive load, such as in first-person shooter gaming. An alternative use proposed by this thesis is *evaluator* think aloud. The same approach is used to help understand an evaluator's thought process during an evaluation. This is particularly useful when trying to apply a novel method they have little experience of, such as playthrough evaluation. 47, 67, 139

usability a quality emerging from a product in use, relating to how it fulfils the users' needs. More detail in Section 2.2 (Usability). 45, *see* user experience, playability & player experience

usability evaluation method any method designed to evaluate the usability of a system or product. These include inspection, expert methods such as heuristic evaluation and cognitive walkthrough, formal models such as Goals, Operators, Methods, Selection rules (GOMS), as well as empirical approaches including user test. vi, 1, 2, 5–7, 11, 12, 20–22, 31, 38–43, 48, 51–53, 60, 61, 65, 66, 70, 71, 73, 74, 76, 77, 97, 99, 110, 115, 132, 144, 147, 150–152, 224–228, *see* usability, usability inspection method, user test, heuristic evaluation, cognitive walkthrough & Goals, Operators, Methods, Selection rules

usability inspection method a method designed to evaluate the usability of a system or product by inspection, rather than by user test. For example, heuristic evaluation. 57, 58, 224, *see* usability evaluation method, usability & user experience

user action framework a framework for analysing usability issues, particularly those of a cognitive nature (Andre, 2000; Andre et al., 2000, 2001, 2003; Capra, 2001; Catanzaro, 2005; Hartson et al., 1999; Keenan et al., 1999; Mahajan, 2003; Mentis and Gay, 2003; Sridharan, 2001). Includes a hierarchical taxonomy of interaction breakdowns that may cause problems. Based on Norman's Theory of Action (Norman, 1986). 37, 38, 53, 54, 116–118, 120, 121, 144, 154, 155, 226, *c.f.* player action framework

user effect the consequence for usability tests of the particular users involved, whether they are evaluators or end-users. Formally defined as "The varied capacities of individual users as defined by their respective expertise and experiences to capture a subset of detectable usability problems of a system, given the particularities of the context of a usability test." (E. L.-C. Law and Hvannberg, 2004a). 47, 155, *c.f.* evaluator effect

user experience "a person's perceptions and responses that result from the use or anticipated use of a product, system or service" (ISO, 2010).
Also includes the user's relationship with the company (Nielsen Norman Group, 2011), and context (Hassenzahl and Tractinsky, 2006). 12, 13, 15, 23, 27, 30, 43, 63, 77, 84–86, 155, 226, *see* usability & player experience

user interface the input / output of the game, interfacing between player and game system. Can include both hardware and software. For example, keyboard, mouse, control sticks, buttons, onscreen menu or head-up display. 21, 37, 99, 122

user test is a usability evaluation method involving empirical testing with users. v, vi, 2, 9, 14–16, 33, 35, 39, 41–43, 45, 47, 49–51, 53, 54, 58, 60, 63, 64, 67–75, 77, 79, 81, 83–87, 93, 97, 103–106, 109, 110, 116, 118, 123, 124, 130–133, 143, 147–149, 155, 193, 223, 224, 227, *see* usability, usability evaluation method, usability inspection method & user experience

wildcard effect "When the power of the findings is too low to use a statistical test, the sample size may be too low to provide a stable estimate of an effect." In cases like these the wildcard effect is due to participants (evaluators, users, etc) "...who are significantly better or worse than average and whose performance in the conditions of the study do not reflect the usability evaluation method but reflect their Wildcard status." Gray and Salzman (1998). This is especially relevant for usability evaluation methods that rely more on informal procedures and subjective interpretation. With a more structured procedure it should be possible to identify when and where participant errors are made, and improve the methodology to provide better support so that the same errors are not made in the future. 47, *c.f.* evaluator effect, user effect & matcher effect

Windows, Icons, Mouse, Pointer the traditional graphical desktop application interface. Most usability evaluation methods are designed for these forms of interface. Compare to the kinds of immersive, embodied interface seen in first-person shooter games. Often the only concession to traditional interface displays are a head-up display overlay, and some form of menu system which is typically "outside" the game mode. 1, 37, 46, 117

Acronyms

DARe Discovery and Analysis REsources. 110

DEVAN DEtailed Video ANalysis. 54

GOMS Goals, Operators, Methods, Selection rules. 227

RPG role-playing game. 22, 81

RTS real-time strategy. 68, 85

List of Equations

List of Figures

List of Tables